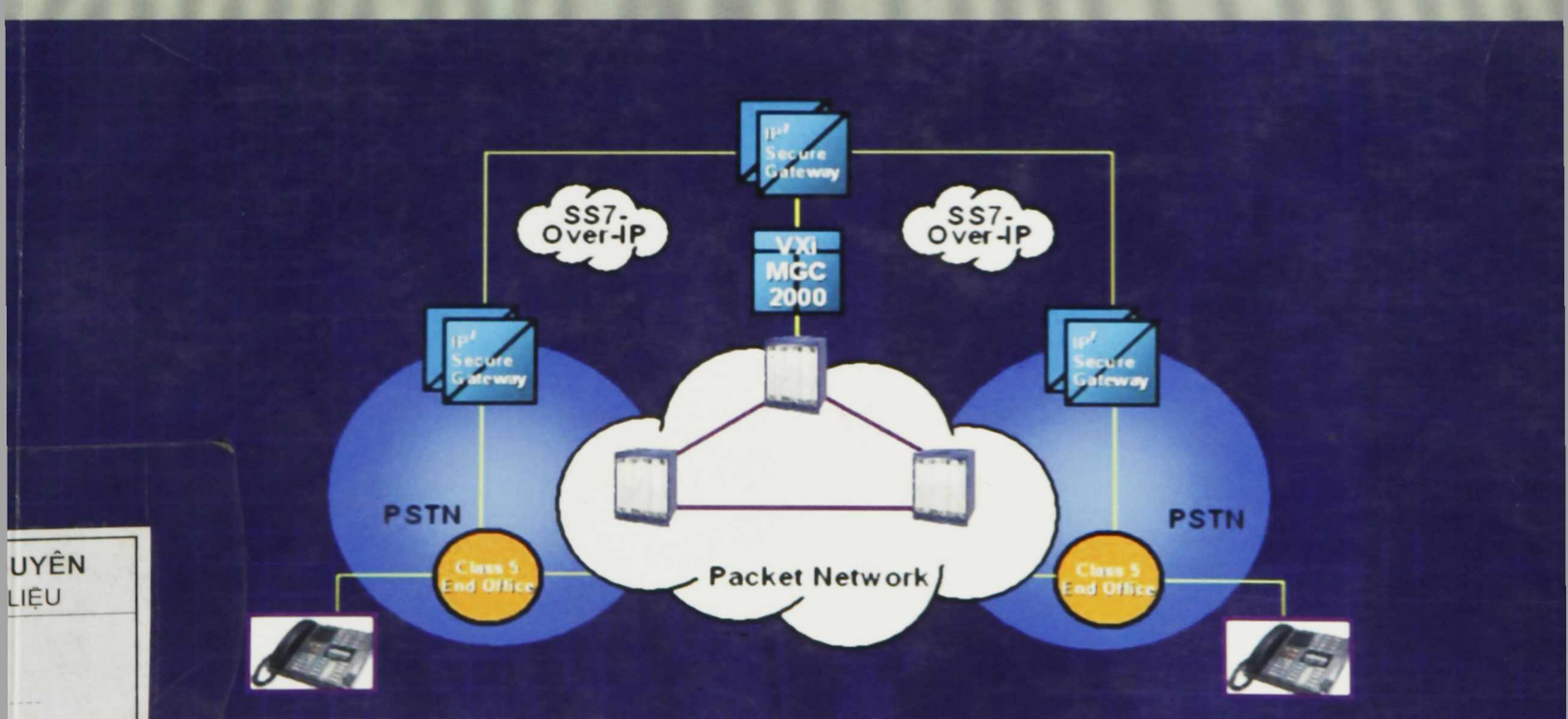
ThS. DƯƠNG VĂN THÀNH

CHUYỂN MẠCH MỀM
VÀ ỨNG DỤNG TRONG MẠNG VIỄN THÔNG THẾ HỆ SAU



UYÊN
LIỆU

NHÀ XUẤT BẢN BƯU ĐIỆN

CHUYỂN MẠCH MỀM

VÀ ỨNG DỤNG TRONG MẠNG VIỄN THÔNG THẾ HỆ SAU

ThS. DƯƠNG VĂN THÀNH

CHUYỂN MẠCH MỀM VÀ ỨNG DỤNG TRONG MẠNG VIỄN THÔNG THẾ HỆ SAU

NHÀ XUẤT BẢN BƯU ĐIỆN
Hà Nội - 2006

Mã số: HV 02 HM 06

LỜI NÓI ĐẦU

Hiện nay, xu hướng hội tụ mạng viễn thông, mạng truyền dữ liệu và mạng máy tính đang diễn ra mạnh mẽ trên thế giới. Mạng viễn thông thế hệ sau (NGN) ra đời để đáp ứng xu hướng đó. "Trái tim" của NGN chính là chuyển mạch mềm. Chuyển mạch mềm đáp ứng cho cung cấp dịch vụ đa phương tiện băng rộng, có các ứng dụng chính hướng tổng đài quá giang gói (Packet Tandem), tổng đài nội hạt, VoIP, mạng thông tin di động; có thể chuyển tải IP quay số ra khỏi mạng điện thoại chuyển mạch công cộng (PSTN); truyền tín hiệu thoại qua chuyển mạch gói; cung cấp các dịch vụ thuần IP; ứng dụng đa phương tiện cho thuê bao IP, có thể cung cấp cho khách hàng các dịch vụ như MM phone C - khi có cuộc gọi đến; dịch vụ đa phương tiện cho phép điện thoại cố định, di động, máy tính... có thể cùng đàm thoại; dịch vụ vừa truy nhập Inlernet vừa đàm thoại trên một đường dây điện thoại; dịch vụ nhắn tin thống nhất cho phép chuyển các cuộc gọi...

*Có thể nói công nghệ chuyển mạch mềm là một trong những vấn đề thời sự hiện nay. Với mục đích giới thiệu cụ thể về chuyển mạch mềm, Nhà xuất bản Bưu điện xuất bản cuốn sách "**Chuyển mạch mềm và ứng dụng trong mạng viễn thông thế hệ sau**" của tác giả ThS. Dương Văn Thành, Học viện Công nghệ Bưu chính Viễn thông. Cuốn sách được trình bày với 8 chương, 2 chương đầu giới thiệu về hiện trạng, xu hướng phát triển của mạng viễn thông và công nghệ thông tin; khái niệm cơ bản về NGN và chuyển mạch mềm. Các chương còn lại cuốn sách sẽ đề cập cụ thể về chuyển mạch mềm như: mô hình tham chiếu chuyển mạch mềm (các mặt phẳng, thực thể chức năng), các giao thức của chuyển mạch mềm (giao thức khởi tạo phiên, giao thức cổng phương tiện MEGACO/H.248, giao thức truyền tải báo hiệu SIGTRAN và một số ứng dụng về chuyển mạch mềm).*

Cuốn sách sẽ là tài liệu tham khảo hữu ích cho các cán bộ và kỹ thuật viên công tác trong lĩnh vực viễn thông; các nhà hoạch định phát triển mạng lưới; các giáo viên,

học sinh nghiên cứu về chuyển mạch mềm, mạng thế hệ sau tại các trường đào tạo chuyên ngành Viễn thông.

Nhà xuất bản Bưu điện mong nhận được ý kiến đóng góp của quý vị và bạn đọc để nâng cao chất lượng trong các lần xuất bản tiếp theo. Ý kiến góp ý xin gửi về ***Nhà xuất bản Bưu điện - 18 Nguyễn Du, Hà Nội.***

Trân trọng cảm ơn./.

Hà Nội, tháng 5 năm 2006

NHÀ XUẤT BẢN BƯU ĐIỆN

Chương 1

TỔNG QUAN VỀ HIỆN TRẠNG VÀ XU HƯỚNG PHÁT TRIỂN CỦA VIỄN THÔNG VÀ CÔNG NGHỆ THÔNG TIN

1.1 SỰ PHÁT TRIỂN CỦA KỸ THUẬT VIỄN THÔNG VÀ MÁY TÍNH

1.1.1 Viễn thông

Ước mơ truyền tin đi xa đã có từ cổ xưa, song chỉ thực sự bắt đầu phát triển mạnh mẽ vào khoảng cuối thế kỷ 18 đầu thế kỷ 19. Ta hãy điểm qua những mốc lịch sử quan trọng nhất của quá trình phát triển kỹ thuật viễn thông:

- Vào khoảng những năm cuối thế kỷ 18 đầu thế kỷ 19 xuất hiện điện báo quang học (Cờ hiệu) ở châu Âu để truyền tin đi xa tới hàng trăm km/giờ. Cho tới khoảng năm 1840 hầu hết các nước châu Âu sử dụng các hệ thống trên nhằm mục đích truyền tin phục vụ cho quân sự và chính phủ.

- Sau khi Michael Farađây phát minh ra các hiện tượng về điện từ, với những tính chất rất quí báu của dòng điện khi truyền dẫn trong các môi trường truyền, các kỹ sư đã cố gắng tìm cách ứng dụng các hiệu ứng điện từ để truyền tin thay cho phương pháp quang học. Người đầu tiên đưa ra ý tưởng và đã thực hiện thành công việc áp dụng các hiện tượng điện để truyền tin là William Sturgeon người Anh dùng nam châm điện (Electromagnet).

- Năm 1822 Francis Ronald đã trình diễn "Điện báo" dựa trên ý tưởng của William Sturgeon. Tuy nhiên ý tưởng này của Francis Ronald không được nhà cầm quyền Anh quốc lúc đó bảo trợ vì khi đó mạng điện báo quang học đang rất thịnh hành tại Anh quốc.

- Năm 1824 tại Niu-oóc giáo sư Samuel Finley Breese Moorse bắt đầu quan tâm tới lĩnh vực điện từ và thông tin điện trên cơ sở ý tưởng của Francis Ronald.

- Năm 1831 Joseph Henry (Mỹ) sáng chế chuông điện từ xa và đó chính là tiền thân của điện báo sau này.

- Năm 1837 dựa trên nguyên lý chuông từ xa Moorse đã trình diễn thành công nguyên tắc của điện báo Mooc trên cơ sở đó mạng điện báo phát triển rất mạnh mẽ và thực sự bắt đầu đánh dấu kỷ nguyên viễn thông.

- Sự kiện rất thú vị là vào khoảng năm 1866 một người Mỹ là Joshua Coppersmit tin rằng có thể làm ra cái mà ông ta gọi là "Telephone" tương tự như điện báo nhưng để truyền tiếng nói đi xa và ông ta đã rất nhiệt huyết truyền bá ý tưởng của mình. Tuy vậy Joshua Coppersmit không bao giờ thực hiện được ý tưởng tuyệt vời này.

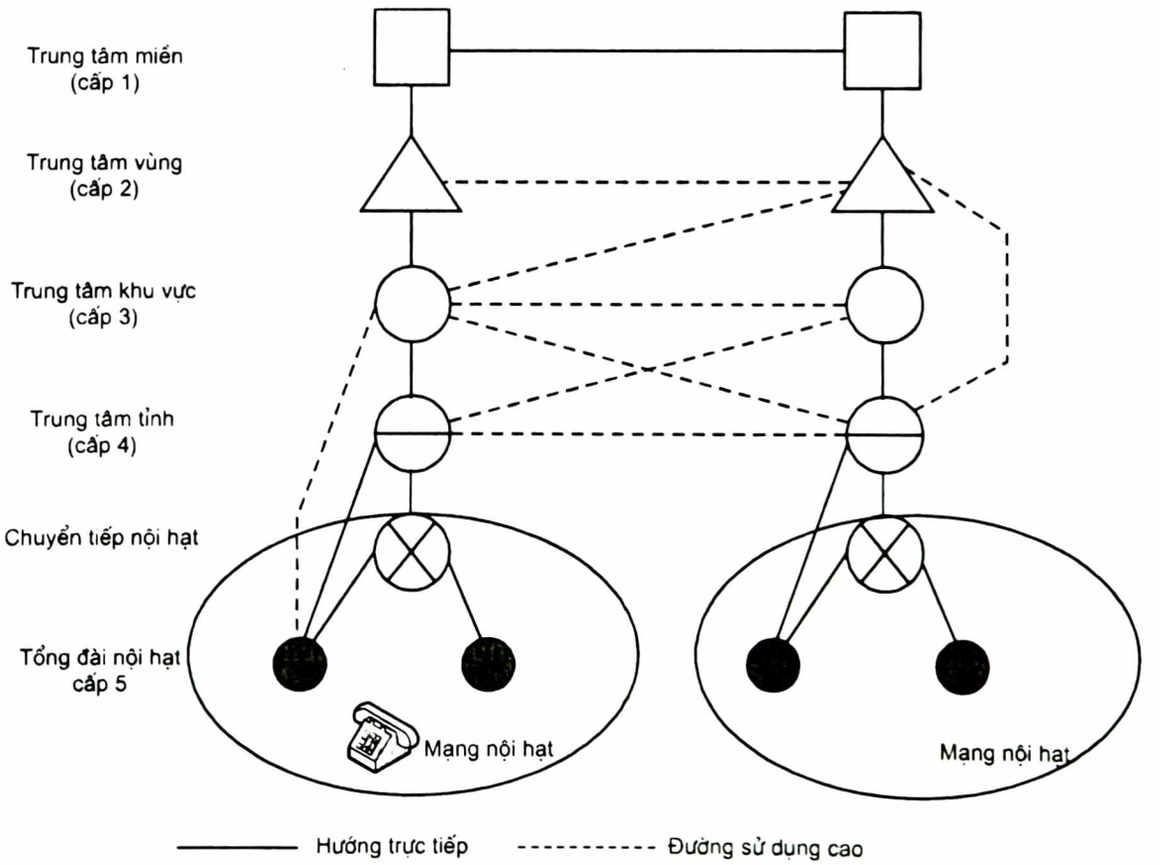
- Năm 1876 Alecxand Graham Bell đã sáng chế và hoàn thiện điện báo nhiều kênh và sau đó sáng chế máy điện thoại thực hiện ý tưởng huyền thoại của Joshua Coppersmit là truyền tiếng nói đi xa. Sáng chế máy điện thoại của Alecxand Graham Bell có ý nghĩa vô cùng trọng đại và sâu sắc đối với xã hội hiện đại, mở ra kỷ nguyên phát triển mạnh mẽ của ngành Viễn thông.

- Năm 1925 John Logie Baird đã thành công truyền bức ảnh truyền hình đen trắng đầu tiên và khoảng 1928 nguyên lý truyền hình màu đã được phát minh ...

Tiếp theo sau sáng chế của Alecxand Graham Bell và John Logie Baird là một loạt các sự kiện quan trọng đánh dấu thời kỳ phát triển mạnh mẽ của kỹ thuật thông tin hiện đại.

- Năm 1877: Máy điện thoại tại nhà riêng đầu tiên được đưa vào sử dụng.
- Năm 1879: Tổng đài nhân công đầu tiên được đưa vào sử dụng.
- Năm 1881: Cuộc gọi điện thoại đường dài đầu tiên được thực hiện.
- Năm 1889: Hai anh em nhà Strowger sáng chế tổng đài tự động hệ từng nấc.
- Năm 1895: Phát minh Vô tuyến điện.
- Năm 1915: Cuộc gọi điện thoại xuyên lục địa đầu tiên.
- Năm 1925: Phát minh ghép kênh theo tần số FDM.
- Năm 1925: Phát minh vô tuyến truyền hình.
- Năm 1929: Phát minh PCM.
- Năm 1947: Phát minh Transistor.
- Năm 1960: Thử nghiệm thông tin vệ tinh.
- Năm 1960: Thử nghiệm trường chuyển mạch điện tử đầu tiên.
- Năm 1963: Sáng chế máy điện thoại ấn phím.
- Năm 1965: Tổng đài điện tử SPC đầu tiên ESS No1 (Mỹ).
- Năm 1970: Tổng đài điện tử số đầu tiên E10A (Pháp).
- Năm 1971: Sáng chế Microprocessor (vi xử lý).
- Năm 1976: Tổng đài điện tử số hoàn toàn đầu tiên DMS100 (Ca-na-da).

- Năm 1977: Sáng chế Internet
- Năm 1977: Sáng chế cáp sợi quang
- Năm 1977: Sáng chế thông tin di động
- Năm 1987: Phát minh mạng thông minh
- Năm 1988: Phát minh ATM và B-ISDN, mở đầu cho công nghệ "Mạng thế hệ sau" NGN (Next Generation Network)
- Năm 1989: Cáp sợi quang đến nhà riêng.
- Năm 1996: Chuyển mạch IP
- Năm 1997: Chuyển mạch mềm
- Năm 2001: Các hãng bắt đầu ứng dụng VoIP
- Năm 2002: Bắt đầu ứng dụng các dịch vụ NGN ...



Hình 1.1: Cấu trúc phân cấp mạng điện thoại công cộng

Cho tới ngày nay, trong kỹ thuật viễn thông thì dịch vụ điện thoại có tỷ trọng lớn và quan trọng nhất do vậy mạng điện thoại công cộng PSTN là mạng được quan tâm

hàng đầu. Các dịch vụ khác như điện báo, truyền số liệu, phát thanh, truyền hình v.v... thường được ghép "ký sinh" trên mạng điện thoại công cộng. Hình 1.1 minh họa cấu trúc tổng quan của mạng PSTN hiện đại điển hình.

Đối với các quốc gia có diện tích lãnh thổ lớn, cấu trúc mạng viễn thông thường được chia thành 5 cấp với các tổng đài chức năng tương ứng tại trung tâm nút mạng. Một tổng đài cụ thể có thể hoạt động kết hợp một số chức năng, ví dụ như tổng đài nội hạt (tổng đài cấp 5) có thể kết hợp cùng với tổng đài chuyển tiếp (transit) nội hạt để tạo thành tổng đài hỗn hợp và được xây lắp tại một mặt bằng trong một nhà chung. Tổng đài cấp 5 có thể kết nối qua tổng đài chuyển tiếp nội hạt hay trực tiếp với tổng đài trung tâm tỉnh (cấp 4). Các tổng đài trung tâm (cấp 4 trở lên) kết nối với các tổng đài cấp cao hơn qua đường trung kế, hiệu suất sử dụng cao nếu như giữa chúng thực sự có nhu cầu liên lạc lớn. Trong những trường hợp này thì trước hết tổng đài thử định tuyến cho các cuộc gọi theo đường sử dụng cao, nếu không còn đường rỗi trong hướng này thì mới định tuyến theo hướng trực tiếp tới tổng đài cấp cao hơn.

Lưu ý rằng ngày nay, với xu hướng sử dụng tổng đài SPC (Stored Programme Control - Điều khiển theo chương trình) trình độ công nghệ cao và cấu trúc phân tán nên số lượng cấp mạng có thể giảm xuống 2-3 mức. Ví dụ chiến lược phát triển mạng viễn thông của ngành Bưu chính Viễn thông đã hoạch định đến năm 2005 mạng viễn thông nước ta cơ bản chuyển sang mạng NGN (Next Generation Network - Mạng thế hệ sau) có cấu trúc 2 mức.

Mạng PSTN (Public Switched Telephone Network - Mạng điện thoại chuyển mạch công cộng) được thiết kế và xây dựng tối ưu cho việc đáp ứng các đặc trưng lưu lượng của dịch vụ điện thoại. Trong bối cảnh hiện nay nhiều hình loại dịch vụ mới phát triển rất mạnh mẽ ví dụ như dịch vụ số liệu, Internet, video v.v... và nhiều động lực kinh doanh của nền kinh tế thị trường toàn cầu hóa, mạng PSTN truyền thống đã bộc lộ nhiều nhược điểm do đó đòi hỏi phải chuyển đổi mạng PSTN hiện có sang mạng thế hệ mới có khả năng đáp ứng tốt mọi yêu cầu cung cấp và ứng dụng dịch vụ. Dưới đây liệt kê một số nhược điểm căn bản nhất:

- Dịch vụ số liệu với các đặc trưng lưu lượng rất khác so với các đặc trưng lưu lượng dịch vụ điện thoại, ví dụ như: băng thông sử dụng, tốc độ truyền bit, tính thời gian thực, chất lượng dịch vụ v.v... sẽ phát triển nhanh và vượt trội trong tương lai gần.

- Mạng PSTN không thể tạo và cung cấp các tính năng dịch vụ mới một cách nhanh chóng, hiệu quả cho người dùng. Với sự cạnh tranh gay gắt của nền kinh tế thị trường ngày càng tăng đòi hỏi việc tạo các dịch vụ mới phải thực hiện nhanh chóng,

mềm dẻo để kịp thời cung cấp cho thị trường, chống sự độc quyền và phụ thuộc vào nhà sản xuất.

• Các dịch vụ số liệu/điện thoại/video rất khó, thậm chí không thể hội tụ trong mạng PSTN hiện có.

• Kiến trúc hạ tầng mạng PSTN dựa trên cơ sở công nghệ TDM (Time Divison Mutiplexing - Ghép kênh theo thời gian) và chuyển mạch kênh không mềm dẻo và không hiệu quả cho việc truyền số liệu và các dịch vụ đa phương tiện (multimedia).

Mạng NGN được xây dựng trên cơ sở công nghệ chuyển mạch gói và IP (Internet Protocol - Giao thức Internet) nên cơ bản đã khắc phục được những nhược điểm nêu trên của mạng PSTN và có các đặc điểm vượt trội về sử dụng băng thông, hiệu năng của mạng v.v...

1.1.2 Máy tính

Có tới hàng trăm kiểu máy tính điện tử số đã được thiết kế và thử nghiệm trong quá trình phát triển để đạt tới trình độ máy tính hiện đại. Một số lớn đã bị lãng quên theo thời gian, nhưng một số ít còn lại đã có ảnh hưởng quan trọng đặc biệt tới tiến trình phát triển của lĩnh vực này. Máy tính hiện đại đã đạt tới những thành tựu đỉnh cao của trí tuệ con người và trong mạng viễn thông hiện đại không thể thiếu vai trò của máy tính. Dưới đây nhắc lại một số thành tựu điển hình.

Thủy tổ của kỹ thuật máy tính phải nói tới nhà toán học lỗi lạc người Pháp Blaire Pascal, người mà hậu thế đã đặt tên cho một ngôn ngữ lập trình bậc cao nổi tiếng Pascal để tỏ lòng tôn kính. Khoảng năm 1642 Pascal đã thiết kế máy tính cơ học có khả năng thực hiện hai phép tính cộng và trừ. 30 năm sau, nhà bác học người Đức Leibniz đã hoàn thiện máy tính của Pascal để thực hiện cả bốn phép tính số học. Trong suốt vài trăm năm sau máy tính không có gì thay đổi đáng kể. Mãi tới khoảng năm 1834 Charler Babbage - giáo sư trường đại học tổng hợp Cambriger sáng chế máy tính số đầu tiên và ông đặt tên cho máy tính này là "Analotical Engine". Đó là máy tính cơ học đa năng và điều khiển bằng phần mềm do Ada Lovelace (con gái của đại thi hào Lord Bairown) lập trình và do vậy mà một ngôn ngữ lập trình hiện đại có tên gọi là Ada để kỷ niệm sự kiện đó.

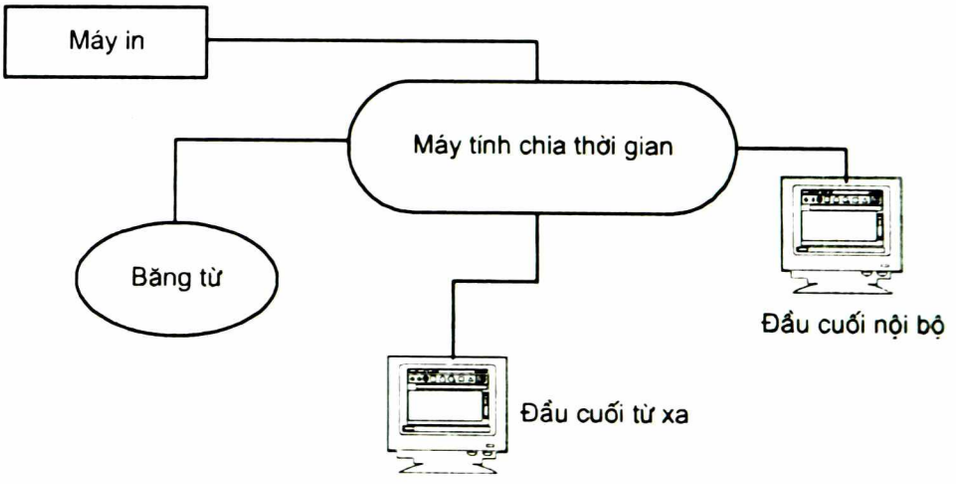
- Năm 1943: Sáng chế máy tính điện tử số đầu tiên COLOSSUS.

- Năm 1946: Sáng chế máy tính điện tử số ENIAC (Electronic Numerical Integrator And Computer) mở đầu cho giai đoạn phát triển mạnh mẽ của ngành máy tính điện tử số. ENIAC chứa tới 18000 đèn điện tử, 1500 rơle, khối lượng tới 30 tấn và tiêu thụ điện năng đến 140 kW.

- Năm 1949: Wilkes sáng chế máy tính điện tử số SPC đầu tiên mang tên EDSAC. Cùng trong khoảng thời gian này, John Von Neumann rất quan tâm lĩnh vực máy tính điện tử số, ông đã đề xuất ý tưởng tập trung hóa chương trình biểu diễn dưới dạng số nhị phân trong bộ nhớ của máy tính cùng với số liệu. Ý tưởng này được sử dụng trong EDSAC năm 1949 và ngày nay hầu hết các máy tính điện tử số được xây dựng theo cấu trúc của EDSAC.

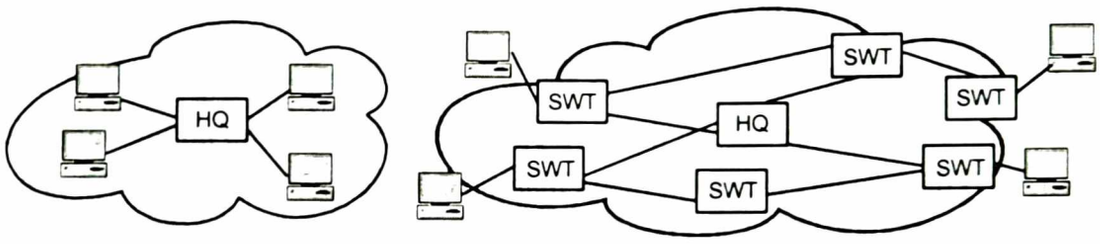
Trong giai đoạn tiếp theo máy tính điện tử số SPC phát triển rất mạnh mẽ và đạt được các thành tựu to lớn. Năm 1963, lần đầu tiên xuất hiện trên thị trường máy tính làm việc với ngôn ngữ bậc cao. Năm 1980 máy tính cá nhân (PC) đầu tiên đã được ứng dụng.

Trong khoảng những năm 1940-1950 máy tính điện tử số ra đời và phát triển mạnh mẽ, tuy nhiên đó là các máy tính đơn lẻ vì khi đó giá thành của một máy tính điện tử số là rất đắt. Đến khoảng những năm 1960 nhằm đáp ứng yêu cầu tăng nhanh các ứng dụng máy tính điện tử số của người dùng trong điều kiện giá thành của máy tính điện tử số còn rất đắt quan điểm "chia thời gian" (Time Sharing) bắt đầu phát triển nhanh, nhờ đó một máy tính có thể chia sẻ cho nhiều người cùng sử dụng đồng thời. Hình 1.2 mô tả hệ thống máy tính ứng dụng nguyên lý chia thời gian, nhờ đó chỉ sử dụng một máy tính nhưng đồng thời có khả năng thực hiện nhiều nhiệm vụ khác nhau, phục vụ cho nhiều người sử dụng.



Hình 1.2: Hệ thống máy tính "chia thời gian"

Đến những năm đầu của thập niên 1970, vi mạch LSI/VLSI/Microprocessor ra đời và phát triển nhanh, nhờ đó máy tính cá nhân PC (Personal Computer) ra đời và bắt đầu phát triển rất mạnh mẽ. Sau đó chúng có nhu cầu kết nối mạng để chia sẻ chung nguồn tài nguyên, lúc đầu theo cấu hình tập trung và sau đó theo cấu hình phân tán như minh họa trên hình 1.3.



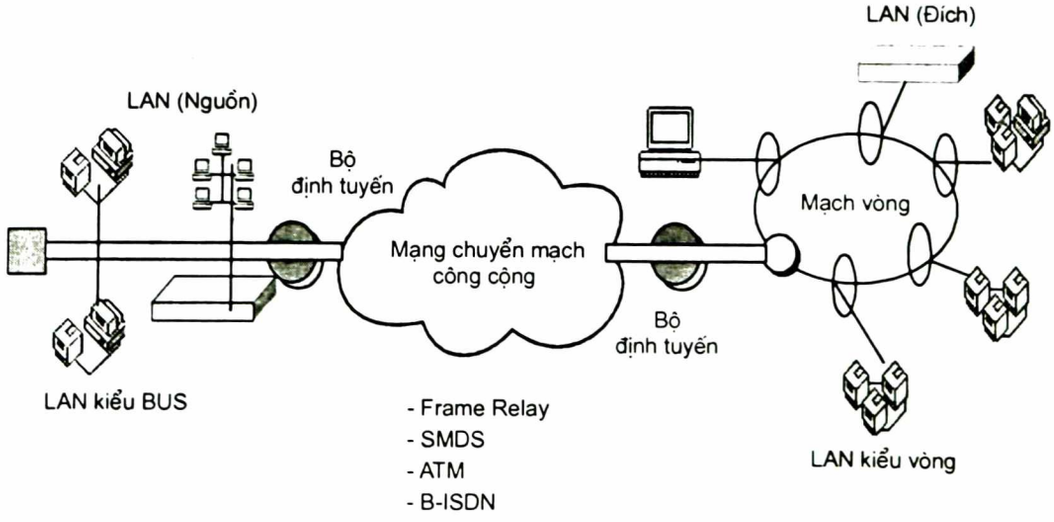
a. Cấu hình tập trung b. Cấu hình phân tán

Chú giải: HQ-Head Quater: Trung tâm; SWT-Service Work-Station: Trạm công tác

Hình 1.3: Cấu hình máy tính tập trung và phân tán

Đầu những năm 1980 kỹ thuật máy tính phát triển rất mạnh mẽ và được ứng dụng rộng rãi trong nhiều lĩnh vực của khoa học và đời sống xã hội, nhu cầu kết nối mạng gia tăng do đó xuất hiện các mạng máy tính cục bộ LAN (Local Area Network). Có hai kiểu cấu hình cơ bản mạng LAN - Cấu hình BUS và cấu hình mạch vòng.

Mạng máy tính phát triển rất nhanh và được ứng dụng rộng rãi do vậy công nghệ LAN phát triển nhanh trong giai đoạn này, nhu cầu liên kết các mạng LAN để mở rộng phạm vi ứng dụng. Các mạng LAN có thể liên kết với nhau qua mạng điện thoại công cộng PSTN như minh họa trên hình 1.4.



Hình 1.4: Các cấu trúc mạng LAN

Tổng quát, ngày nay việc liên kết mạng máy tính nói chung có thể thực hiện theo bốn kiểu kiến trúc mạng thường gặp gồm:

- Cấu hình BUS
- Cấu hình mạch vòng

- Cấu hình phân mảnh
- Cấu hình chuyển mạch

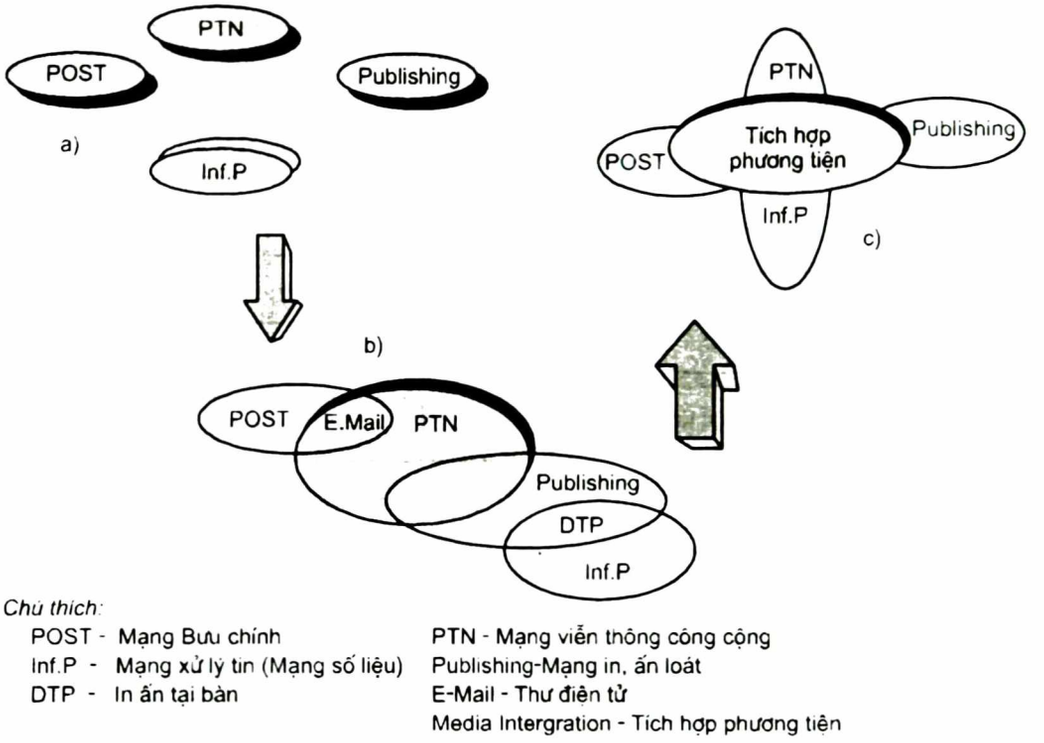
Sự ra đời và phát triển của kỹ thuật và công nghệ bán dẫn vào những năm đầu của thập niên 1960 đã thúc đẩy một cách mạng mạnh mẽ nhiều ngành kinh tế và khoa học kỹ thuật, trong đó có ngành Viễn thông và Công nghệ thông tin. Tuy nhiên do có sự khác biệt tương đối về bản chất của tín hiệu sử dụng cho các dịch vụ điện thoại, điện báo, truyền thông số liệu... và các hạn chế về kỹ thuật lúc đó mà các mạng viễn thông và xử lý tin được xây dựng và hoạt động một cách tương đối độc lập với nhau (hình 1.5). Trong tiến trình phát triển không ngừng của công nghệ viễn thông và công nghệ xử lý tin, đặc biệt là sự phát triển và khả năng ứng dụng tuyệt vời của công nghệ số bao gồm kỹ thuật bán dẫn và vi mạch, kỹ thuật số và kỹ thuật vi xử lý đã làm xuất hiện và thúc đẩy mạnh mẽ khả năng kết hợp các dịch vụ thông tin khác nhau vào một mạng thống nhất - Mạng viễn thông số đa dịch vụ tích hợp ISDN với tốc độ kênh cơ sở là 64 kbit/s (kênh B) sử dụng cho dịch vụ điện thoại, truyền số liệu,... hay 16 kbit/s (kênh D) sử dụng cho báo hiệu DSS1 trên cùng đoạn thuê bao, CCS7 trên mạng trung kế và truyền số liệu chuyển mạch gói, 384, 1536 hay 1920 kbit/s (kênh H) dùng cho các dịch vụ tốc độ cao. Các kiểu kênh cơ bản nêu trên được nhóm lại dưới dạng các cấu trúc kênh 2B+D, 23B+D, 30B+D với các tốc độ dòng bit 144, 1544 và 2048 kbit/s tương ứng. Sử dụng các kênh cơ bản và các cấu trúc nhóm kênh như mô tả trên đây, mạng thông tin đa dịch vụ ISDN băng hẹp đã đáp ứng được nhiều hình loại dịch vụ thông tin truyền thống đang được ứng dụng khá phổ biến trong các nước có nền kinh tế phát triển.

1.2 XU HƯỚNG HỘI TỤ CÔNG NGHỆ VÀ TÍCH HỢP DỊCH VỤ

Tiến trình phát triển của kỹ thuật viễn thông và kỹ thuật xử lý tin được biểu diễn và mô tả sinh động như trên hình 1.5. Từ bốn nhóm dịch vụ truyền thống đơn lẻ (Hình 1.5a, quá khứ) phát triển xuất hiện nhu cầu tích hợp và hội tụ dịch vụ và công nghệ. Trình độ kết hợp từ đơn giản từng cặp mức thấp (hình 1.5b, hiện tại) tiến tới tích hợp và hội tụ toàn phần (hình 1.5c, tương lai).

Tuy nhiên, sự phát triển mạnh mẽ của công nghệ thông tin và nhu cầu trao đổi thông tin ngày càng tăng nhanh và đa dạng hóa của xã hội đòi hỏi phải bổ sung cấp bách các dịch vụ truyền ảnh bao gồm cả ảnh tĩnh và ảnh động chất lượng cao và truyền số liệu, truyền tệp tin tốc độ siêu cao mà chúng yêu cầu tốc độ bit tới vài trăm Mbit/s, thậm chí tới hàng chục Gbit/s. Nói chung mạng ISDN băng hẹp không thể đáp ứng thỏa mãn được các yêu cầu bổ sung nêu trên. Một mạng viễn thông thống nhất đáp ứng tất cả các hình loại dịch vụ viễn thông và xử lý tin với tốc độ yêu cầu rất khác nhau từ một vài

kbit/s đến hàng chục Gbit/s, thậm chí hàng Tbit/s gọi là mạng số liên kết đa dịch vụ băng rộng B-ISDN (Broadband ISDN) và chỉ có B-ISDN mới có khả năng cung cấp các dịch vụ đa phương tiện (Multimedia Services).



Chú thích:

POST - Mạng Bưu chính
Inf.P - Mạng xử lý tin (Mạng số liệu)
DTP - In ấn tại bàn
PTN - Mạng viễn thông công cộng
Publishing-Mạng in, ấn loát
E-Mail - Thư điện tử
Media Intergration - Tích hợp phương tiện

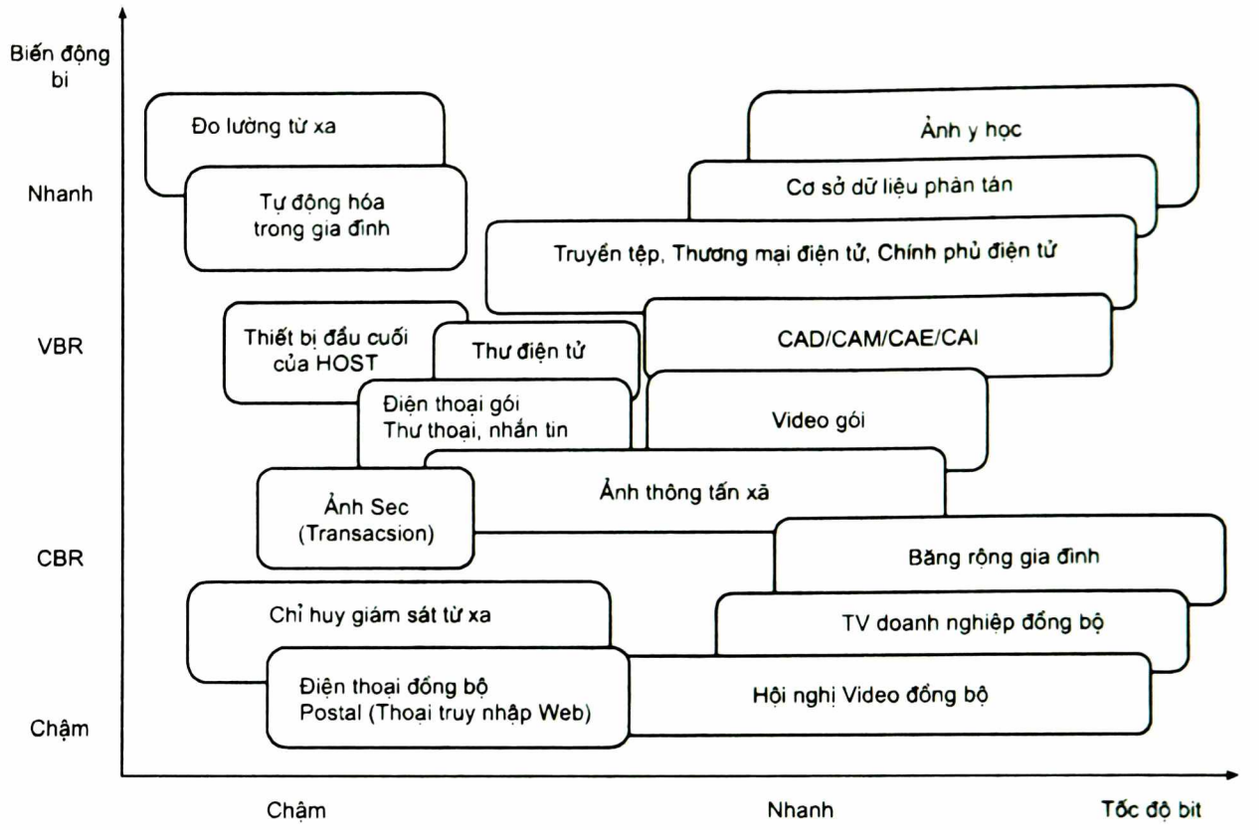
Hình 1.5: Quá trình phát triển của mạng viễn thông và xử lý tin

1.3 CÁC DỊCH VỤ VIỄN THÔNG CƠ BẢN

Hình 1.6 biểu diễn đặc tính thời gian cơ bản của các ứng dụng dịch vụ viễn thông và công nghệ thông tin hiện đại.

Mục tiêu của B-ISDN là nhằm bao gồm các dịch vụ mà chúng sẽ được cung cấp bởi mạng viễn thông và mạng số liệu trong tương lai. Đó không chỉ bao gồm các dịch vụ băng hẹp như điện thoại, số liệu tốc độ thấp, dịch vụ Fax, máy in từ xa, đo lường điều khiển từ xa, teletex, videotex và bưu chính điện tử mà còn bao gồm cả các dịch vụ băng rộng như điện thoại thấy hình, truyền số liệu tốc độ cao, Faximine màu, CATV (truyền hình cáp), HDTV (truyền hình có độ phân dải cao), phân phối âm thanh chất lượng cao, thư video, giám sát video, ảnh độ phân giải cao, hội nghị video, videotex băng rộng v.v... Các dịch vụ B-ISDN cũng có thể phân loại tương ứng với hướng truyền tải tin, quan hệ thời gian và kiểu kênh phân loại dịch vụ kết nối.

Các dịch vụ B-ISDN có thể chia thành các dịch vụ tương tác (interactive services) và dịch vụ phân tán (disributed services) phụ thuộc vào hướng truyền tin.



Chú thích:

VBR: Tốc độ bit thay đổi CAM: Sản xuất được hỗ trợ bằng máy tính CAE: Kỹ thuật hỗ trợ bằng máy tính

CBR: Tốc độ bit cố định CAD: Thiết kế bằng máy tính CAI: Hướng dẫn được hỗ trợ bằng máy tính

Hình 1.6 Phân bố các dịch vụ B-ISDN

Các dịch vụ tương tác lại có thể phân thành loại các dịch vụ đối thoại, dịch vụ bản tin và các dịch vụ hồi phục (Retrieval Services), còn các dịch vụ phân tán có thể phân loại thành các dịch vụ điều khiển được và các dịch vụ không điều khiển được.

Bảng 1.1 Phân loại các dịch vụ B-ISDN

Loại chính	Loại	Các ví dụ ứng dụng
Các dịch vụ tương tác	Các dịch vụ hội thoại	Điện thoại thấy hình
		Hội nghị truyền hình
	Các dịch vụ bản tin thông báo	MHS, thư truyền hình
	Các dịch vụ phục hồi	Phim, tài liệu văn bản
Các dịch vụ phân phối	Các dịch vụ không điều khiển được	Truyền hình, phát thanh
	Các dịch vụ điều khiển được	Ảnh video đồ họa

Dịch vụ hội thoại

Nhìn chung, các dịch vụ hội thoại là kiểu dịch vụ đảm bảo thời gian thực và liên lạc song hướng giữa những thuê bao tham gia quá trình liên lạc. Trung bình, luồng (lưu lượng) tin dịch vụ được truyền tải đối xứng theo hai hướng nhưng nó đôi khi cùng có thể là không đối xứng. Ví dụ về hình loại dịch vụ hội thoại là điện thoại thoại thấy hình, hội nghị video,... Bảng 1.2 giới thiệu các dịch vụ hội thoại.

Bảng 1.2: Các dịch vụ hội thoại

Kiểu tin	Ví dụ dịch vụ	Ứng dụng
Ảnh động	Điện thoại thấy hình băng rộng	Đào tạo từ xa, mua hàng từ xa
	Hội nghị truyền hình băng rộng	Đào tạo từ xa, mua hàng từ xa
		Quảng cáo từ xa
	Giám sát video	Giám sát an toàn xây dựng, giám sát giao thông
	Thông tin âm thanh và video	Thông tin đối thoại TV, video/âm thanh
Âm thanh	Truyền thanh đa chương trình	Truyền thanh đa chương trình
		Kênh truyền thanh nhiều thứ tiếng
Truyền số liệu	Truyền số liệu tốc độ cao	LAN/WAN, Truyền ảnh tĩnh
		CAD/CAM tương tác đa điểm
	Truyền tệp lớn	Truyền tệp
	Tương tác tốc độ cao	Điều khiển thời gian thực, cảnh báo
		Đo lường từ xa
	Fax tốc độ cao	Truyền văn bản, ảnh, bản vẽ
Tài liệu	Truyền ảnh độ phân giải cao	Các ảnh chuyên nghiệp, ảnh y học
		Trò chơi từ xa
	Dịch vụ thông tin tư liệu	Truyền tư liệu hỗn hợp

THƯ VIỆN ĐẠI HỌC THÁI NGUYÊN

Dịch vụ bản tin

Dịch vụ bản tin đảm bảo thông tin liên lạc cho các thuê bao ở hai đầu cuối bằng các thiết bị nhớ. Các thiết bị nhớ này có thể lưu trữ, chuyển tải và xử lý các bản tin nghĩa là nó xử lý có khả năng biên dịch, xử lý và biến đổi thông tin. Ví dụ về dịch vụ bản tin bao gồm truyền gửi phim kiểu bưu chính, ảnh độ phân giải cao, thư thoại và dịch vụ xử lý bản tin MHS (Message Handling Service).

Bảng 1.3 Dịch vụ bản tin

Kiểu tin	Ví dụ dịch vụ	Ứng dụng
Ảnh động và âm thanh	Dịch vụ thư truyền hình	Hộp thư truyền hình và thoại điện tử
Tài liệu	Dịch vụ thư tài liệu	Hộp thư điện tử cho tài liệu hỗn hợp
		Thương mại điện tử

Dịch vụ hồi phục (Retrieval service)

Dịch vụ hồi phục bao gồm khôi phục thông tin công cộng được nhớ trong trang Web thông tin (ngân hàng tin), khi có yêu cầu, thông tin mong muốn được truyền tải tới người dùng vào một cơ sở dữ liệu cá nhân. Trong trường hợp này thời gian khởi đầu của dãy thứ tự thông tin có thể được điều khiển bởi người dùng. Các vùng đích của dịch vụ phục hồi bao gồm phim, ảnh có độ phân giải và chất lượng cao, hộp thư thoại và thông tin lưu trữ.

Bảng 1.4 Dịch vụ Phục hồi

Kiểu tin	Ví dụ dịch vụ	Ứng dụng
Văn bản, số liệu, đồ họa, âm thanh, ảnh tĩnh, ảnh động	Videotex băng rộng	Videotex bao gồm cả ảnh động, đào tạo từ xa, phần mềm từ xa, mua hàng từ xa, quảng cáo từ xa, báo chí điện tử
	Các dịch vụ phục hồi video	Giải trí, đào tạo từ xa
	Các dịch vụ phục hồi ảnh độ phân giải cao	Giải trí, đào tạo từ xa, ảnh chuyên nghiệp, truyền thông ảnh y học
	Các dịch vụ phục hồi tư liệu	Phục hồi các tư liệu hỗn hợp
	Dịch vụ phục hồi số liệu	Phần mềm từ xa

Dịch vụ phân phối không điều khiển

Dịch vụ phân phối không điều khiển giới thiệu kiểu dịch vụ quảng bá chung. Tiêu biểu đó là dòng liên tục thông tin được truyền bởi nhà cung cấp dịch vụ trung tâm tới một số lượng không hạn chế người dùng trong phạm vi mạng. Trong trường hợp này người dùng truy nhập tới dòng tin nhưng không điều khiển được thời gian khởi luồng đầu hay chương trình của luồng tin. Ví dụ điển hình của dịch vụ không điều khiển đó là phát thanh, truyền hình quảng bá.

Dịch vụ phân phối điều khiển được

Các dịch vụ phân phối điều khiển được tương tự như hình loại phân phối không điều khiển được dưới khía cạnh là nhà cung cấp dịch vụ trung tâm cung cấp luồng liên tục thông tin tới nhiều người dùng. Nhưng không giống kiểu dịch vụ phân phối không điều khiển được, ở đây luồng tin cấu thành từ các gói tin mà nó có tích lặp lại theo chu kỳ. Tiếp theo đó, người dùng có thể lựa chọn trong các gói tin phân phối theo chu kỳ và như vậy có thể điều khiển thời gian khởi đầu và nội dung chương trình của các gói tin.

Nói cách khác, nhờ tính có chu kỳ của nội dung tin, người dùng luôn luôn có thể thu được các gói tin được chọn từ đầu đến cuối một cách trọn vẹn.

Ví dụ của hình loại dịch vụ này là videography (Ảnh đồ hoạ) kênh quảng bá.

Bảng 1.5: Dịch vụ Phân phối

Kiểu tin	Ví dụ dịch vụ	Ứng dụng
A. Các dịch vụ không điều khiển video	NTSC, PAL, SECAM TV EDTV, HDTV TV Trả tiền ngay khi xem	Các chương trình truyền hình Các chương trình truyền hình Các chương trình truyền hình
- Văn bản, đồ họa, số liệu Số liệu	Các dịch vụ phân phối tài liệu Phân phối tin tốc độ cao	Báo chí điện tử, xuất bản điện tử Phân phối số liệu không hạn chế
- Ảnh động và âm thanh	Video, âm thanh	Phân phối video - âm thanh
B. Các dịch vụ điều khiển được văn bản, đồ họa, âm thanh, ảnh tĩnh	Kênh video đồ họa quảng bá	Đào tạo từ xa, phần mềm từ xa, mua hàng từ xa, quảng cáo từ xa, báo chí điện tử phục hồi

Ngoài cách phân loại dịch vụ như đã trình bày trên đây có thể còn có nhiều cách phân loại khác. Ví dụ một cách chia dịch vụ thành ba nhóm như sau:

1. Các dịch vụ thông tin: Thoại, âm thanh, hình ảnh, bản tin, hội nghị, mạng riêng...

2. Các dịch vụ nội dung: Audio/Video, bản tin... truyền đơn hướng tới người dùng.

3. Dịch vụ giao dịch (Transactional services): Các giao dịch tài chính, buôn bán lẻ, thương mại điện tử, trình duyệt Web...

Chương 2

TỔNG QUAN VỀ MẠNG THẾ HỆ SAU VÀ CHUYỂN MẠCH MỀM

2.1 CẤU HÌNH TỔNG QUAN MẠNG THẾ HỆ SAU

2.1.1 Sự ra đời của mạng thế hệ sau

Nhu cầu trao đổi thông tin phản ánh trình độ phát triển của xã hội. Khi thông tin được thừa nhận như một nguồn tài nguyên quý giá, nhu cầu về trao đổi thông tin của con người ngày càng đòi hỏi cao về chất lượng, đa dạng về loại hình thông tin và chủng loại dịch vụ. Điều này tạo ra những cơ hội về doanh thu cho những nhà cung cấp dịch vụ viễn thông, nhưng cũng đặt ra không ít những khó khăn về mặt công nghệ.

Trước đây, lưu lượng chủ yếu là tín hiệu thoại, một hạ tầng cơ sở viễn thông theo mạng điện thoại công cộng PSTN cũng đủ đáp ứng nhu cầu khách hàng. Mạng PSTN hoạt động trên cơ sở chuyển giao theo chế độ kênh (Circuit Mode) với những tổng đài chuyển mạch kênh cho phép chuyển mạch tín hiệu thoại với độ tin cậy cao, đảm bảo rất tốt tính thời gian thực.

Ngày nay do sự tác động của hai yếu tố: sự gia tăng nhu cầu của khách hàng và sự ra đời của những công nghệ mới, hạ tầng viễn thông của mỗi nước đang đứng trước những bước ngoặt. Sự gia tăng nhanh nhu cầu của khách hàng về loại hình dịch vụ, không chỉ đơn thuần dịch vụ điện thoại mà còn có nhiều dịch vụ mới, bao gồm cả hình ảnh, dữ liệu và các dịch vụ đa phương tiện... Nếu như lưu lượng thoại được đáp ứng rất tốt bởi mạng PSTN truyền thống thì với những loại lưu lượng còn lại nhưng mạng PSTN lại tỏ ra có rất nhiều nhược điểm:

- Sử dụng băng tần không linh hoạt
- Lãng phí tài nguyên hệ thống
- Không có cơ chế phát hiện và sửa lỗi
- Hiệu năng sử dụng mạng thấp
- Khó hội tụ và tích hợp dịch vụ

... như

Để thỏa mãn nhu cầu của khách hàng, đồng nghĩa với việc gia tăng lợi nhuận, các nhà cung cấp dịch vụ viễn thông yêu cầu những giải pháp công nghệ mới thay thế (hay bổ sung) cho mạng PSTN. Cùng với sự gia tăng nhu cầu của khách hàng, công nghệ chuyển mạch gói cũng góp phần đưa ngành công nghiệp viễn thông chuyển sang thời kỳ mới. Công nghệ chuyển mạch gói đưa ra giải pháp chuyển giao thông tin dưới dạng các gói tin theo phương thức hướng kết nối (connection oriented) hay không kết nối (connectionless) trên các kênh ảo (chỉ thực sự chiếm dụng tài nguyên khi có lưu lượng trên nó). Mạng chuyển mạch gói có thể được xây dựng trên các giao thức khác nhau: X25, IP... trong đó giao thức IP đang là giao thức được quan tâm nhiều nhất. Mạng chuyển mạch gói dựa trên giao thức IP được coi là giải pháp công nghệ đáp ứng sự gia tăng nhu cầu của khách hàng. Với khả năng của mình, các dạng lưu lượng khác nhau được xử lý hoàn toàn trong suốt trong mạng IP, điều này cho phép mạng IP có khả năng cung cấp các loại dịch vụ đa dạng, phong phú bao gồm cả dịch vụ đa phương tiện chứ không riêng gì dịch vụ thoại. Điều này rất có ý nghĩa khi trong tương lai, thông tin thoại chỉ còn tồn tại như dịch vụ giá trị gia tăng.

Như vậy, để đáp ứng nhu cầu khách hàng các nhà quản trị mạng có hai sự lựa chọn: hoặc xây dựng một cơ sở hạ tầng hoàn toàn mới cho mạng IP hoặc xây dựng một mạng có khả năng cung cấp các dịch vụ IP bằng cách nâng cấp trên cơ sở mạng PSTN hiện có.

Trên quan điểm kinh tế, rõ ràng phương án hai là sự lựa chọn đúng đắn, nghĩa là cần phải giải quyết nhiệm vụ chuyển đổi một cách hợp lý quá trình chuyển đổi từ mạng PSTN truyền thống trên cơ sở công nghệ TDM và chuyển mạch kênh sang mạng mạng thế hệ sau NGN trên cơ sở chuyển mạch gói và công nghệ IP.

2.1.2 Các đặc điểm và ưu điểm của mạng thế hệ sau

Hai đặc điểm quan trọng nhất của mạng thế hệ sau NGN đó là: Mạng tích hợp đa dịch vụ và phát triển trên cơ sở hạ tầng viễn thông sẵn có với kiến trúc mở và công nghệ chuyển mạch gói:

- Các lớp chức năng được tích hợp theo chiều ngang, nghĩa là cùng chung một cơ sở hạ tầng thực hiện hội tụ các chức năng mạng trên lớp truyền dẫn chung dựa trên cơ sở công nghệ chuyển mạch gói và được chia sẻ bởi các dịch vụ khác nhau.

- Lớp điều khiển được tách độc lập với lớp truyền dẫn và có khả năng cung cấp các giao diện lập trình mở (API) nhằm đảm bảo môi trường tạo dịch vụ mới một cách mềm dẻo, đa năng và hiệu quả.

• Kiến trúc mạng như trên sẽ đem lại nhiều lợi ích với các năng lực đầy hứa hẹn:

• Nhờ sự độc lập giữa chức năng truyền dẫn và điều khiển kết nối, việc cung cấp dịch vụ mới chỉ đơn giản là việc bổ sung thêm các server (máy chủ) vào lớp dịch vụ nằm phía trên lớp truyền dẫn.

• Lớp điều khiển dịch vụ độc lập với lớp truyền dẫn cũng làm giảm thiểu ảnh hưởng của việc ứng dụng các công nghệ truyền dẫn mới.

• Tất cả các loại hình dịch vụ đều có thể chia sẻ chung một mạng lõi, lưu lượng thoại và dữ liệu không cần phải phân biệt.

• Có khả năng cung cấp các dịch vụ đa phương tiện.

2.1.3 Mô hình kiến trúc mạng thế hệ sau

Xu hướng chung hiện nay là hình thành và xây dựng các mạng NGN thông qua việc chuyển đổi mạng PSTN trên nền cơ sở hạ tầng chuyển mạch kênh TDM sang cơ sở hạ tầng chuyển mạch gói và công nghệ IP. Tuy nhiên trong quá trình chuyển đổi, cần một giải pháp lai ghép giữa hai phương thức chuyển giao thông tin theo kênh và theo gói. Điều này được đáp ứng bằng cách sử dụng các cổng phương tiện (Media Gateway - MG) có cấu trúc phân tán, dưới sự điều khiển của thiết bị điều khiển cổng phương tiện (Media Gateway Controller - MGC).

Hình 2.1 trình bày mô hình tổng quát mạng NGN với việc ứng dụng công nghệ chuyển mạch mềm.

Từ hình vẽ mô tả mô hình mạng thế hệ sau chúng ta thấy rõ rằng trước hết, mạng NGN tổng thể được chia thành hai mạng thành phần - Mạng truyền dẫn gói tốc độ cao và mạng điều khiển tương thích với kiến trúc hệ thống mở hiện đại.

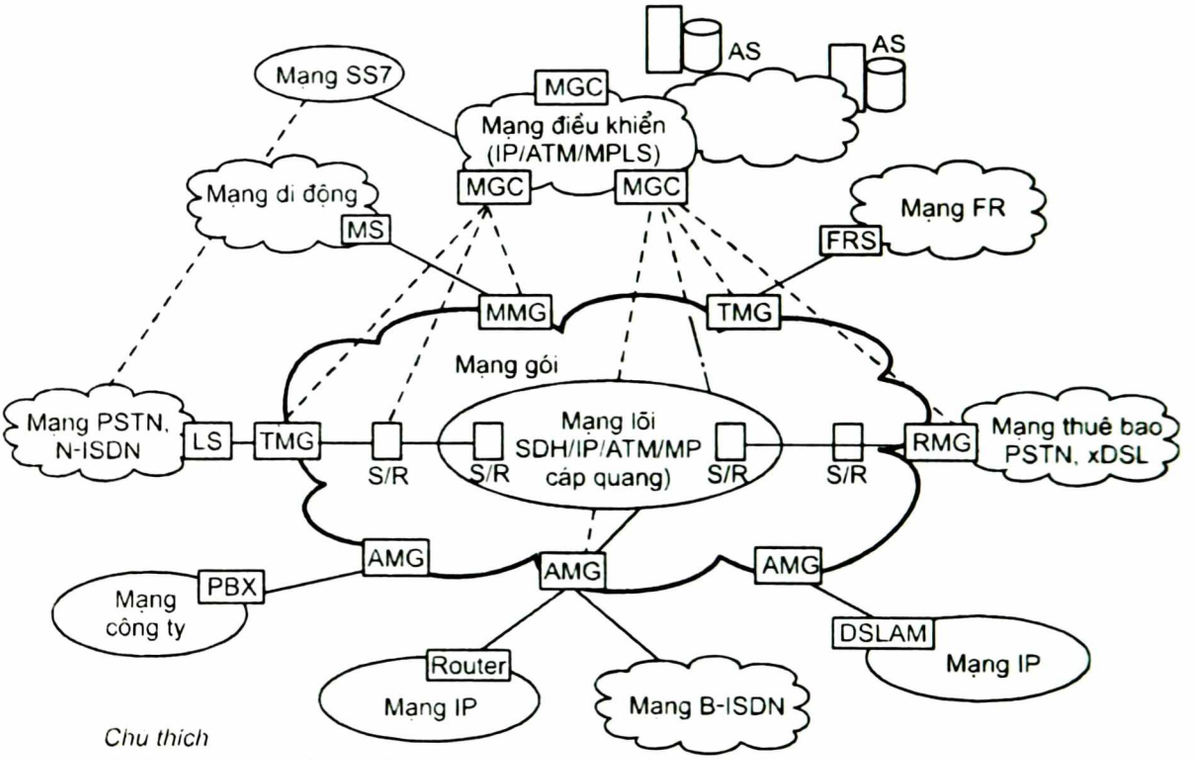
Mạng truyền dẫn gói tốc độ cao có cấu trúc phân cấp bao gồm hai thành phần cơ bản là mạng lõi và mạng truyền tải gói.

Mạng lõi cấu thành từ mạng cáp quang tốc độ cao sử dụng các công nghệ truyền dẫn SONET/SDH, WDM, ATM, MPLS và IP cùng với các thiết bị chuyển mạch - định tuyến SR (Switch/Router) dung lượng lớn.

Mạng truy cập bao gồm các mô-đun cổng kết nối (Gateway) như AMG, MMG, RMG, TMG, các hệ thống chuyển mạch nhánh (Edge switching system) và mạng truyền dẫn trên cơ sở chuyển mạch gói tốc độ cao.

Cổng RMG dùng để kết nối trực tiếp các thuê bao của mạng PSTN/xDSL. Cổng AMG có thể trang bị các giao diện PRI, V5.x để kết nối với các thiết bị tập trung như

tổng đài PBX, cáp sợi quang FLC của mạng công ty. AMG cũng có thể kết nối trực tiếp với các bộ định tuyến của mạng số liệu nói chung, trong đó bao gồm mạng IP để cung cấp các dịch vụ Internet cho khách hàng. Các cổng trung kế TMG thực hiện các chức năng của tổng đài đường dài/chuyển tiếp để kết nối với các mạng khác như mạng PSTN, N-ISDN hay mạng chuyển tiếp khung FR... qua giao diện NNI. Cổng MMG hỗ trợ cho sự tích hợp mạng để kết nối với mạng thông tin di động.



Chu thích

AMG	Access Media Gateway	Cổng phương tiện truy nhập
AS	Application Server	Máy chủ ứng dụng
MGC	Media Gateway Controller	Điều khiển cổng phương tiện
MMG	Mobile Media Gateway	Cổng phương tiện di động
MG	Media Gateway	Cổng phương tiện
RMG	Residential Media Gateway	Cổng phương tiện nội hạt
TMG	Trunk Media Gateway	Cổng phương tiện trung kế
SS7	Signalling System No7	Báo hiệu số 7
FR	Frame Relay	Chuyển tiếp khung
FRS	Frame Relay Switch	Chuyển mạch chuyển tiếp khung
MS	Mobile Switch System	Hệ thống chuyển mạch di động
LS	Local System	Hệ thống nội hạt
DSLAM	Digital Subscriber Line Access Multip.	Ghép kênh truy nhập đường dây thuê bao số
IP	Internet Protocol	Giao thức Internet
S/R	Switch/ Router	Chuyển mạch/Bộ định tuyến

Hình 2.1: Mô hình mạng NGN với công nghệ chuyển mạch mềm

Mạng điều khiển là mạng liên kết các máy chủ ứng dụng (AS) và các tổng đài cổng MGC. AS để điều khiển và cung cấp các ứng dụng và dịch vụ cho người dùng còn MGC để điều khiển các cổng MG và các tài nguyên mạng khác. MGC còn có các tên gọi khác là chuyển mạch mềm (Softswitch), máy chủ xử lý cuộc gọi hoặc Call Agent. Chuyển mạch mềm là hệ thống phần mềm điều khiển phân tán đảm bảo khả năng điều khiển cuộc gọi và xử lý báo hiệu cho mạng thế hệ sau. Tại mức cơ bản nhất, chuyển mạch mềm phải cung cấp các tính năng điều khiển cuộc gọi liên quan tới thiết lập và giải phóng cuộc nối, định tuyến cuộc gọi, lựa chọn các thuộc tính dịch vụ bao gồm dịch vụ điện thoại, video, Internet, dịch vụ đa phương tiện..., xác định thẩm quyền cuộc gọi, quản lý tối ưu nguồn tài nguyên và chất lượng dịch vụ theo yêu cầu. Hiện nay chuyển mạch mềm chú trọng thực hiện các tính năng chủ yếu sau:

- Điều khiển phân tán và thông minh các cuộc kết nối tại các cổng MG, đảm bảo tính độc lập và tính chuyển đổi thích ứng cao đối với các phương tiện kỹ thuật sử dụng cụ thể khác nhau như ATM, IP, TDM, SCN (mạng chuyển mạch kênh) và PDN (mạng số liệu gói).

- Định tuyến cuộc gọi trong phạm vi mạng dựa trên cơ sở báo hiệu và các thông tin trong cơ sở dữ liệu của thuê bao.

- Có khả năng chọn các tiến trình áp dụng cho xử lý cuộc gọi của các hình loại dịch vụ khác nhau bao gồm điện thoại, âm thanh - hình ảnh, Internet/Web và đa phương tiện. Hỗ trợ đa giao thức điều khiển và báo hiệu, các chức năng quản lý như xử lý lỗi, tính cước v.v...

- Có khả năng chuyển điều khiển cuộc gọi cho các thành phần của mạng khác. Tương tác liên mạng mềm dẻo và hiệu năng cao.

- Giao diện và hỗ trợ các chức năng mã hóa - giải mã, nén - Giải nén tín hiệu dịch vụ nhằm giảm băng thông và tối ưu hóa sử dụng tài nguyên mạng, đáp ứng chất lượng dịch vụ QoS (Quality of Service) theo yêu cầu của khách hàng và hiệu năng của mạng.

Các giải pháp công nghệ và kiến trúc hạ tầng mạng NGN mới dựa trên cơ sở mạng chuyển mạch gói tốc độ cao, công nghệ SDH/SONET, IP/ATM/MPLS và phần mềm điều khiển phân tán tạo điều kiện cho sự hội tụ các giao thức truyền thông, chuyển mạch, định tuyến và xử lý tín hiệu dịch vụ đảm bảo sự tích hợp các dịch vụ viễn thông bao gồm điện thoại, âm thanh, video, số liệu, Internet... với chất lượng dịch vụ QoS theo yêu cầu của người sử dụng là những vấn đề đang được quan tâm, nghiên cứu để ứng dụng vào thực tiễn.

Trong một vài năm gần đây trên phạm vi thế giới công nghệ chuyển mạch mềm bắt đầu được đặc biệt quan tâm nghiên cứu và đã đạt được các thành tựu áp dụng vào mạng các tổng đài trung kế đạt kết quả tốt.

2.2 CÁC KHÁI NIỆM VỀ CHUYỂN MẠCH MỀM

Các ý kiến khác nhau về chuyển mạch mềm cũng xuất phát từ góc độ nhìn nhận khác nhau về kiến trúc, chức năng và các đặc trưng của mạng. Trước khi đi tới một khái niệm chung, có thể tham khảo một số quan điểm về chuyển mạch mềm của một số hãng khác nhau.

CommWorks: (www.commworks.com). Chuyển mạch mềm (Softswitch) bao gồm các mô-đun phần mềm tiêu chuẩn, có chức năng điều khiển cuộc gọi, báo hiệu, có giao thức liên kết và khả năng thích ứng với các dịch vụ mới trong mạng hội tụ. Thêm vào đó, chuyển mạch mềm thực hiện chuyển mạch cuộc gọi mà không phụ thuộc vào phương thức truyền dẫn cũng như cách truy nhập mạng, các dạng lưu lượng khác nhau được xử lý trong suốt. Thông qua mạng IP, chuyển mạch mềm cung cấp các dịch vụ IP với các yêu cầu ngày càng cao của khách hàng.

MobileIN: (www.mobileIN.com). Chuyển mạch mềm là khái niệm trong đó bao hàm việc tách phần cứng mạng ra khỏi phần mềm mạng. Trong mạng chuyển mạch kênh truyền thống, phần cứng và phần mềm của hệ thống chuyển mạch không độc lập với nhau mà là thực thể toàn vẹn của một nhà cung cấp. Mạng chuyển mạch kênh dựa trên những thiết bị chuyên dụng cho việc kết nối và được thiết kế với mục đích phục vụ thông tin thoại. Những mạng chuyển mạch gói với hiệu năng cao hơn sẽ sử dụng giao thức IP để định tuyến thông tin thoại và số liệu qua các tuyến khả dụng và các thiết bị dùng chung.

Alcatel: (www.alcatel.com) với sản phẩm 5424Softswitch sử dụng để giảm tải Internet và chạy các ứng dụng VoIP H.323, hệ thống 1000 Softswitch ứng dụng làm tổng đài chuyển tiếp gói. Chuyển mạch mềm là trung tâm điều khiển trong cấu trúc mạng viễn thông. Nó cung cấp khả năng chuyển tải thông tin một cách mềm dẻo, an toàn và đáp ứng các đặc tính mong đợi khác của mạng. Đó là các sản phẩm có chức năng quản lý dịch vụ, điều khiển cuộc gọi gatekeeper, thể hiện ở việc hội tụ các công nghệ IP, ATM, TDM trên nền cơ sở hạ tầng sẵn có. Hơn nữa, 1000 softswitch còn có khả năng tương thích giữa chức năng điều khiển cuộc gọi và các chức năng mới sẽ phát triển sau.

Như vậy, tùy thuộc vào thị trường của mình, các nhà cung cấp khác nhau có quan điểm khác nhau về chuyển mạch mềm, tuy nhiên các quan điểm đó bổ sung cho nhau

để hình thành một định nghĩa chung về chuyển mạch mềm. Chuyển mạch mềm là hệ thống chuyển mạch thực hiện đầy đủ chức năng của chuyển mạch truyền thống, có khả năng kết hợp nhiều loại dịch vụ, có thể đáp ứng nhiều loại lưu lượng, khả năng kết nối với nhiều loại mạng, nhiều loại thiết bị, dễ dàng nâng cấp cũng như tương thích với các dịch vụ mới và các dịch vụ trong tương lai.

Tổ chức chuyển mạch mềm quốc tế ISC (International Softswitch Consortium) và Nhóm đặc trách kỹ thuật Internet IETF (Internet Enginerring Task Force) đều hướng tới chuyển mạch mềm với kiến trúc mạng phân tán với các thành phần chức năng hoàn toàn độc lập nhau - các thành phần chức năng bao gồm truyền tải, chuyển mạch, điều khiển mạng và các lô-gíc dịch vụ. Lợi ích mang lại là mạng này không gặp phải sự giới hạn về phần cứng như trong mạng truyền thống (đó là sự cần thiết giữa chuyển mạch kênh và truyền tải, giữa các dịch vụ mạng thông minh với cơ chế và lô-gíc dịch vụ). Hệ thống chuyển mạch mềm hoàn toàn là chuyển mạch trong đó các thực thể chức năng tồn tại trong các thiết bị thành phần vật lý khác nhau và phân tán. Hiện nay chức năng truyền dẫn đang bắt đầu chuyển cho các thành phần mạng trên cơ sở công nghệ IP/ATM/MPLS. Trong tương lai, chức năng điều khiển mạng, lô-gíc dịch vụ cũng sẽ tách rời khỏi chức năng chuyển mạch. Việc phân bố các thực thể chức năng sẽ cho phép dễ dàng phát triển các thuộc tính tiên tiến và chuyển giao với chi phí thấp. Nỗ lực tách các thực thể chức năng ra khỏi chuyển mạch làm cho chức năng chuyển mạch trở nên đơn giản hơn, hiệu quả hơn và rẻ hơn. Các lô-gíc dịch vụ phân bố cũng làm cho việc phát triển các ứng dụng không bị hạn chế bởi các dịch vụ chuyển giao, điều khiển và môi trường tạo dịch vụ tập trung.

Mạng NGN (Hình 2.1) được xây dựng trên cơ sở ứng dụng giải pháp công nghệ chuyển mạch mềm sẽ có cấu hình mạng lõi là các thiết bị chuyển mạch-định tuyến SR tốc độ cao, dung lượng lớn và chúng được liên kết bằng mạng chuyển mạch gói trên cơ sở các công nghệ IP, ATM/ MPLS. Mạng truyền tải gói là các thiết bị chuyển mạch MG. Mạng truyền tải giao tiếp với các mạng ngoài thông qua các MG hoạt động dưới sự điều khiển của MGC. Phần mạng truy cập là các nút truy nhập băng rộng và thiết bị truy nhập tích hợp IAD.

Chương 3

MÔ HÌNH THAM CHIẾU CHUYỂN MẠCH MỀM

Để có một khái niệm toàn diện và thống nhất về chuyển mạch mềm, tổ chức chuyển mạch mềm quốc tế ISC (International Softswitch Consortium) đã đưa ra mô hình tham khảo của chuyển mạch mềm với các thực thể và mặt phẳng chức năng.

3.1 CÁC MẶT PHẲNG CHỨC NĂNG

Các mặt phẳng chức năng được đưa ra theo nghĩa chung nhất nhằm phân biệt các thực thể chức năng trong kiến trúc chuyển mạch mềm. ISC đưa ra 4 mặt phẳng chức năng: mặt phẳng truyền tải (Transport plane); mặt phẳng điều khiển và báo hiệu (Call control and Signaling plane); mặt phẳng ứng dụng và dịch vụ (Service and Application plane); mặt phẳng quản lý (Management plane).

3.1.1 Mặt phẳng truyền tải

Mặt phẳng truyền tải thực hiện chức năng vận chuyển các bản tin giữa các thực thể trong toàn mạng. Các bản tin có thể là bản tin báo hiệu cuộc gọi, bản tin thiết lập cuộc gọi hay lưu lượng cần truyền. Cơ chế truyền tải các bản tin này có thể dựa trên bất kỳ một công nghệ truyền dẫn nào, miễn là nó đáp ứng các yêu cầu của việc truyền dẫn loại lưu lượng đó.

Mặt phẳng truyền tải cũng đưa ra các điểm truy nhập dành cho việc liên kết với các mạng khác nhằm mục đích báo hiệu hay truyền tải lưu lượng giữa chúng. Các thực thể và chức năng của mặt phẳng này được điều khiển bởi các chức năng trong mặt phẳng báo hiệu và điều khiển cuộc gọi.

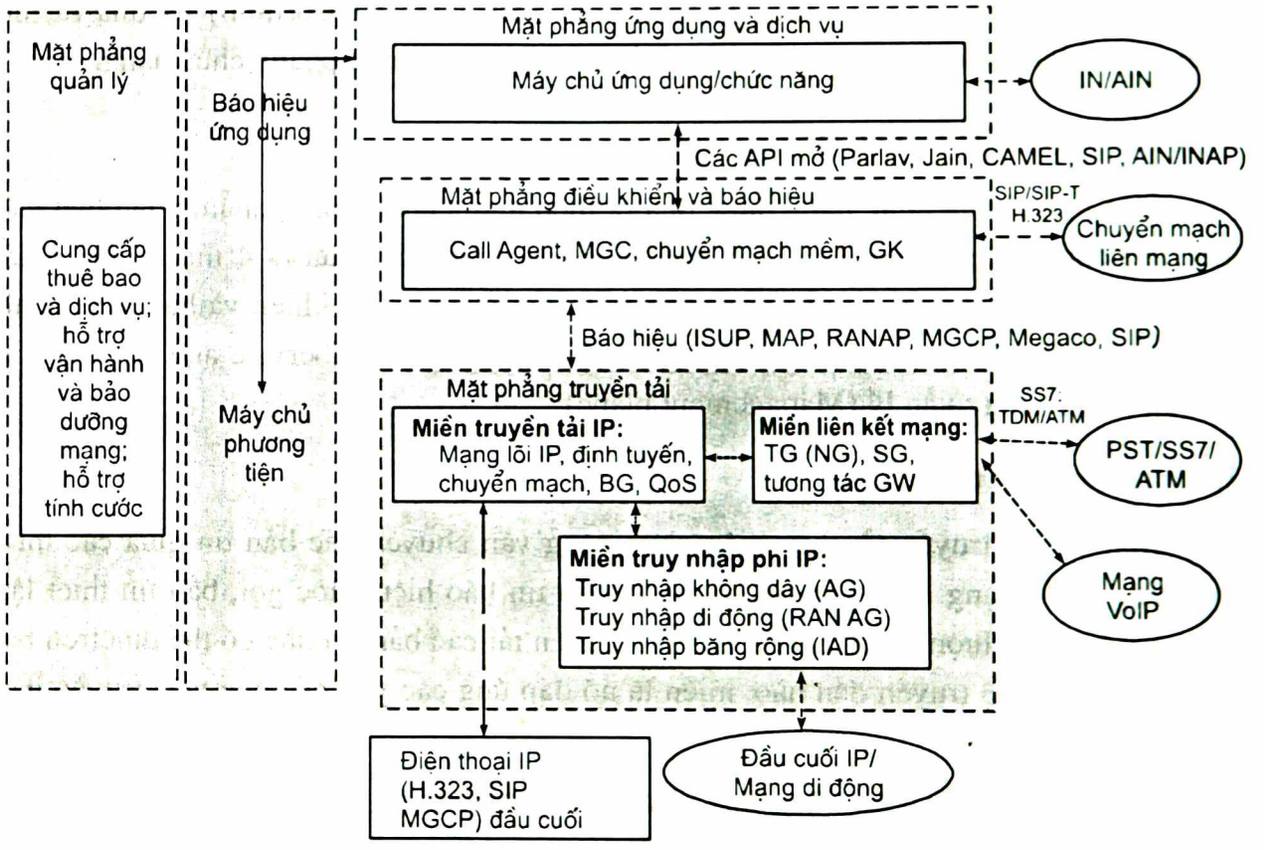
Mặt phẳng truyền tải cũng có thể được chia nhỏ thành 3 miền:

• Miền truyền tải IP: Miền này cung cấp hệ thống lõi và chuyển mạch, định tuyến cho việc truyền tải các bản tin qua mạng IP. Miền truyền tải IP bao gồm các thiết bị: định tuyến, chuyển mạch, thiết bị cung cấp cơ chế giám sát chất lượng dịch vụ hay các chính sách truyền tải.

• Miền liên kết mạng: Miền này đảm nhận chức năng chuyển đổi các dạng lưu lượng và báo hiệu giữa các mạng khác nhau nhằm đảm bảo sự tương thích giữa chúng,

bao gồm các thiết bị: cổng báo hiệu (Signaling Gateway), cổng phương tiện (Media Gateway), cổng liên kết mạng (Interworking Gateway).

• Miền truy nhập không IP: Miền này nhằm hỗ trợ việc truyền dẫn giữa mạng IP và các đầu cuối không IP hay mạng vô tuyến. Vùng này bao gồm các thiết bị: cổng truy nhập (Access Gateway) hay các cổng nội hạt (Residential Gateway), đầu cuối ISDN, thiết bị truy nhập tích hợp IAD, các cổng phương tiện cho các mạng di động hay mạng truy nhập vô tuyến...



Hình 3.1: Mô hình tham chiếu chuyển mạch mềm ISC

3.1.2 Mặt phẳng báo hiệu và điều khiển cuộc gọi

Mặt phẳng này điều khiển hầu hết các thực thể của mạng, đặc biệt là các thực thể thuộc mặt phẳng truyền tải. Các thiết bị và chức năng trong mặt phẳng này tiến hành điều khiển cuộc gọi dựa trên các bản tin báo hiệu nhận được từ mặt phẳng truyền tải, nó cũng thực hiện việc thiết lập và giải phóng cuộc gọi trên cơ sở điều khiển các thiết bị thuộc mặt phẳng truyền tải. Mặt phẳng này bao gồm các thiết bị như: bộ điều khiển cổng phương tiện (Media Gateway Controller), Gatekeeper (thiết bị điều khiển cổng kết nối), máy chủ LDAP.

3.1.3 Mặt phẳng ứng dụng và dịch vụ

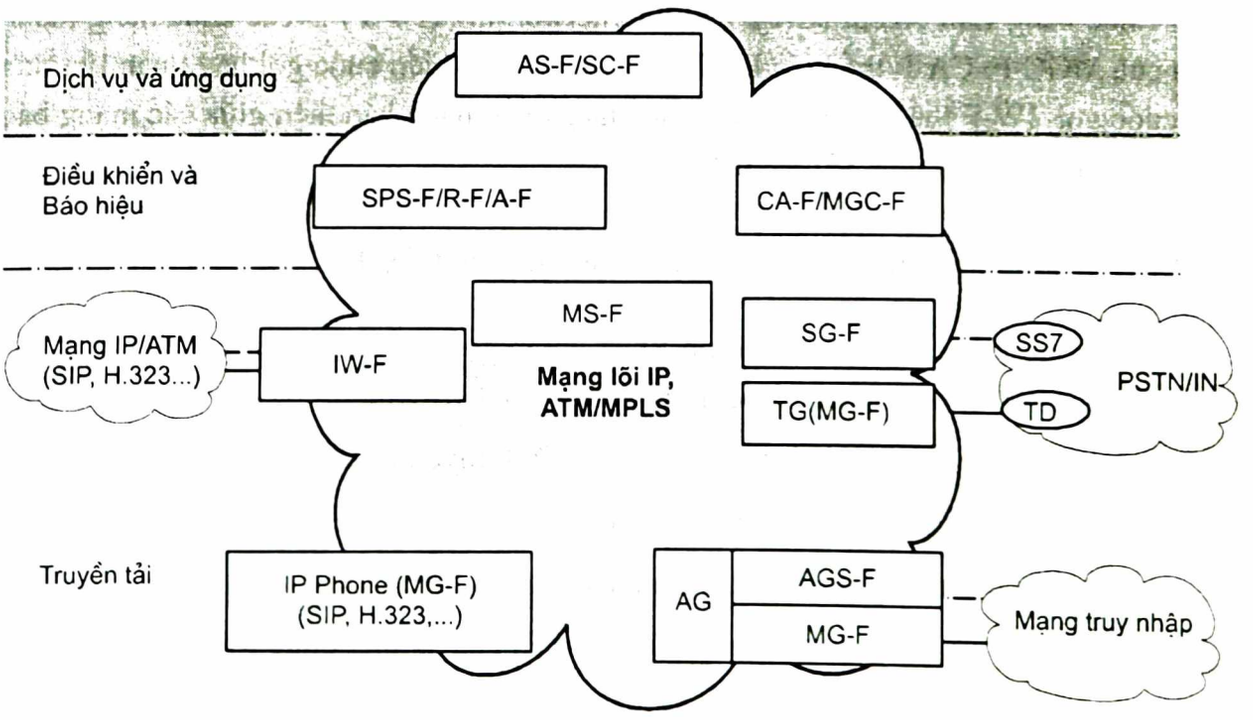
Mặt phẳng này cung cấp các lô-gíc dịch vụ hay kịch bản của các dịch vụ và ứng dụng trên mạng. Các thiết bị trong mặt phẳng sẽ điều khiển từng bước cuộc gọi dựa trên các lô-gíc có sẵn thông qua việc giao tiếp với các thiết bị trong mặt phẳng báo hiệu và điều khiển cuộc gọi. Mặt phẳng này bao gồm các thiết bị như máy chủ ứng dụng (Application Servers), máy chủ chức năng (Feature Servers).

3.1.4 Mặt phẳng quản lý và bảo dưỡng mạng

Mặt phẳng này thực hiện các chức năng quản lý như tính cước, hỗ trợ vận hành, các xử lý liên quan tới thuê bao hay cung cấp dịch vụ tới khách hàng. Mặt phẳng quản lý có thể tương tác với 3 mặt phẳng trên thông qua các giao diện chuẩn hay giao diện lập trình mở.

3.2 CÁC THỰC THỂ CHỨC NĂNG

Kiến trúc của mạng khi triển khai chuyển mạch mềm có thể được tham khảo qua hình 3.2. Mỗi khối chức năng trên có thể thực hiện một chức năng nhưng cũng có thể kết hợp cùng thực hiện một chức năng. Theo đó các thực thể chức năng thuộc các mặt phẳng chức năng khác nhau.



Hình 3.2: Các thực thể chức năng chuyển mạch mềm

3.2.1 Chức năng điều khiển cổng phương tiện (MGC-F)

Chức năng này thường được thực hiện bởi thực thể vật lý MGC (Media Gateway Controller). Đây là một trong những thiết bị quan trọng nhất và được biết tới với nhiều tên như Call Agent, Call controller hay chuyển mạch mềm. Chức năng MGC-F cung cấp lô-gíc dịch vụ và báo hiệu điều khiển cuộc gọi cho MG. MGC-F có các đặc điểm:

- Duy trì trạng thái cuộc gọi đối với mọi cuộc gọi MG
- Điều khiển giao tiếp giữa các MG cũng như giữa MG với các thiết bị đầu cuối
- Đóng vai trò là trung gian thoả thuận các tham số kết nối giữa các đầu cuối thuộc các MG
- Tiếp nhận và khởi tạo các bản tin báo hiệu đi và tới các điểm kết cuối và các mạng bên ngoài
- Tương tác với máy chủ ứng dụng nhằm cung cấp các dịch vụ tới khách hàng
- Quản lý một số tài nguyên mạng như: các cổng MG, băng thông...
- Giao tiếp với các chức năng đinh tuyến và tính cước để hỗ trợ cho việc tính cước, nhận thực và định tuyến
- Có thể tham gia vào nhiệm vụ quản lý trong môi trường mạng di động
- Chức năng này gồm các giao thức ứng dụng Megaco/H.248 và MGCP

Chức năng Call Agent (CA-F) và Internetworking (IW-F) là các chức năng thành phần của MGC-F. CA-F thể hiện khi MGC xử lý điều khiển cuộc gọi hay quản lý trạng thái cuộc gọi. IW-F thể hiện khi MGC thực hiện chức năng báo hiệu giữa các mạng báo hiệu khác nhau (ví dụ như SS7 và SIP).

3.2.2 Chức năng định tuyến cuộc gọi và tính cước (R-F, A-F)

R-F cung cấp các thông tin định tuyến cuộc gọi cho MGC-F, A-F thu thập các thông tin phục vụ cho việc tính cước. R-F và A-F có đặc điểm:

- Cung cấp chức năng định tuyến cho việc định tuyến cuộc gọi liên mạng
- Cung cấp khả năng quản lý phiên và di động
- Cập nhật các thông tin định tuyến từ các nguồn bên ngoài
- Tương tác với AS-F nhằm cung cấp các dịch vụ hay ứng dụng tới khách hàng
- R-F và A-F là thành phần của SPS-F và thường được tích hợp trong chức năng điều khiển cổng MGC-F.

3.2.3 Chức năng cổng báo hiệu và báo hiệu cổng truy nhập (SG-F/AGS-F)

SG-F cung cấp cổng phương tiện cho việc báo hiệu giữa mạng IP và PLMN, PSTN (thường là báo hiệu số 7). Vai trò chính của SG-F là đóng gói và truyền các bản tin báo hiệu số 7 của PSTN (ISUP hoặc INAP) hay PLMN (MAP hoặc CAP) qua mạng IP.

AGS-F cung cấp cổng phương tiện cho việc báo hiệu giữa mạng IP và mạng truy cập dựa trên chuyển mạch kênh. Vai trò chính của AGS-F là đóng gói và truyền các bản tin báo hiệu V5 hay ISDN, BSSAP, RANAP qua mạng IP.

Các đặc điểm của SG-F:

- Đóng gói và truyền các bản tin báo hiệu của mạng PSTN (SS7) (sử dụng giao thức SIGTRAN) tới các MGC-F hay một SG-F khác

- Một SG-F có thể phục vụ nhiều MGC-F

- Khi SG-F và MGC-F không được cài đặt chung, SG-F sẽ thực hiện chức năng giao diện giao thức (ví dụ như SIGTRAN)

- Các giao thức ứng dụng của chức năng này bao gồm: SIGTRAN, TUA, SUA hay M3UA trên SCTP

Các đặc điểm của AGS-F:

- Đóng gói và truyền các bản tin báo hiệu V5 hoặc ISDN (ví dụ như SS7) tới MGC-F

- Một MGC-F có thể phục vụ nhiều AGS-F

- Khi AGS-F và MGC-F không được cài đặt chung, AGS-F sẽ thực hiện chức năng giao thức giao diện (ví dụ như SIGTRAN)

- Các giao thức ứng dụng của chức năng này bao gồm: SIGTRAN, IUA, V5UA hay M3UA trên SCTP.

3.2.4 Chức năng máy chủ ứng dụng (AS-F)

Chức năng chính của AS là cung cấp các lô-gíc dịch vụ ứng dụng. Các đặc điểm của AS-F bao gồm:

- Có thể yêu cầu MGC kết thúc cuộc gọi hay phiên liên lạc để phục vụ cho một ứng dụng nào đó, ví dụ như thư thoại, lập cầu hội nghị...

- Có thể yêu cầu MGC khởi tạo lại các đặc trưng cuộc gọi ví dụ như tính năng chuyển cuộc gọi, chứng thực thẻ gọi trả trước...

- Có thể thay đổi các mô tả lưu lượng thông qua giao thức SDP.

- Có thể điều khiển MS-F nhằm chức năng xử lý lưu lượng.

- Có thể kết nối tới các ứng dụng Web và có các giao diện Web.

- Có giao diện lập trình ứng dụng cho việc tạo các dịch vụ mới.

- Giao tiếp với MGC-F hay MS-F.

- Có thể sử dụng các dịch vụ của MGC-F để điều khiển các nguồn tài nguyên bên ngoài.

- Các giao thức ứng dụng bao gồm: SIP, MGCP, MEGACO/H248, LDAP, HTTP, CPL, XML.

- Các giao diện lập trình ứng dụng mở bao gồm: JAIN và Parlay.

Thông thường, sự kết hợp giữa AS-F và MGC-F sẽ tạo ra các năng lực điều khiển các dịch vụ tăng cường như: điện thoại hội nghị, chờ cuộc gọi... Các nhà khai thác sẽ không sử dụng một giao diện giữa AS và MGC, thay vào đó là một giao diện lập trình ứng dụng API giữa MGC và AS. Khi ấy AS còn có tên gọi khác là máy chủ chức năng.

Chức năng điều khiển dịch vụ SC-F (Service Control Function) xuất hiện khi AS-F điều khiển lô-gíc dịch vụ. SC-F gồm INAP, CAP, MAP, giao diện API mở.

3.2.5 Chức năng cổng phương tiện (MG-F)

MG giao tiếp với mạng IP bằng các đường điểm truy nhập hay trung kế mạng. Nói khác đi, MG-F hoạt động như một cổng giao tiếp giữa mạng IP và các mạng bên ngoài, đó có thể là mạng PSTN hay PLMN v.v... MG-F có thể cung cấp các cổng giao tiếp giữa mạng IP và mạng chuyển mạch kênh hay giữa các mạng chuyển mạch gói với nhau (IP và 3G hay ATM). Vai trò cơ bản của MG-F là: chuyển lưu lượng từ một khung dạng truyền dẫn này sang một khung dạng truyền dẫn khác, thường là giữa chuyển mạch kênh và chuyển mạch gói, giữa gói ATM và gói IP, hoặc giữa các kênh tương tự/ ISDN và các gói giống như trong cổng phương tiện nội hạt.

Các đặc điểm chính của MG-F:

- Luôn luôn có mối quan hệ chủ/tớ với MGC-F thông qua giao thức MGCP hay MEGACO/H.248

- Có thể thực hiện các chức năng xử lý lưu lượng như chuyển mã, đóng gói, loại bỏ tiếng vọng, giám sát hiện tượng rung pha (Jitter), đưa ra các xử lý khi bị mất gói tin ...

- Có thể thực hiện các chức năng chèn lưu lượng như tạo âm báo tiến trình cuộc gọi, tạo âm DTMF, tạo tạp âm xen...

- Thực hiện chức năng báo hiệu và phát hiện sự thay đổi trạng thái của các thiết bị đầu cuối như: phát hiện DTMF, phát hiện sự kiện on/off-hook (đặt/nhấc máy), phát hiện tín hiệu thoại tích cực...

- Quản lý và phân bổ tài nguyên cho các chức năng trên.

- Phân tích các con số nhận được từ thiết bị đầu cuối dựa trên kế hoạch đánh số và quay số do MGC gửi tới.

- Cung cấp cơ chế cho chức năng MGC-F để kiểm tra sự thay đổi trạng thái và năng lực của các điểm kết cuối. MG-F không yêu cầu khả năng quản lý trạng thái các cuộc gọi. MG-F chỉ quản lý trạng thái kết nối của các cuộc gọi mà nó hỗ trợ.

- Máy điện thoại SIP phone kết hợp MG-F và MGC-F trong một thiết bị.

- Cổng ứng dụng SIP kết hợp MG-F và MGC-F trong một thiết bị.

- Các giao thức ứng dụng bao gồm RTP/RTCP, TDM, H248, MGCP.

3.2.6 Chức năng máy chủ phương tiện (MS-F)

Chức năng của máy chủ phương tiện là đáp ứng các yêu cầu của AS-F và MGC-F để thực hiện việc xử lý lưu lượng trên các dòng lưu lượng đóng gói.

Các đặc trưng chính của MG-F:

- Hỗ trợ nhiều bộ mã và chuyển đổi mã
- Hỗ trợ cho việc điều khiển bởi nhiều AS-F hay MGC-F
- Hỗ trợ cho nhiều tính năng đồng thời gồm:
  - o Phát hiện số
  - o Tạo dòng âm báo và thông báo
  - o Tạo âm báo
  - o Ghi, lưu trữ dòng đa phương tiện
  - o Nhận dạng tiếng nói
  - o Tạo tiếng nói từ văn bản
  - o Hỗ trợ tính năng cuộc gọi Hội nghị
  - o Xử lý Fax
  - o Phát hiện thoại tích cực và khoảng trống ...
- Các giao thức ứng dụng bao gồm SIP, H248, MGCP

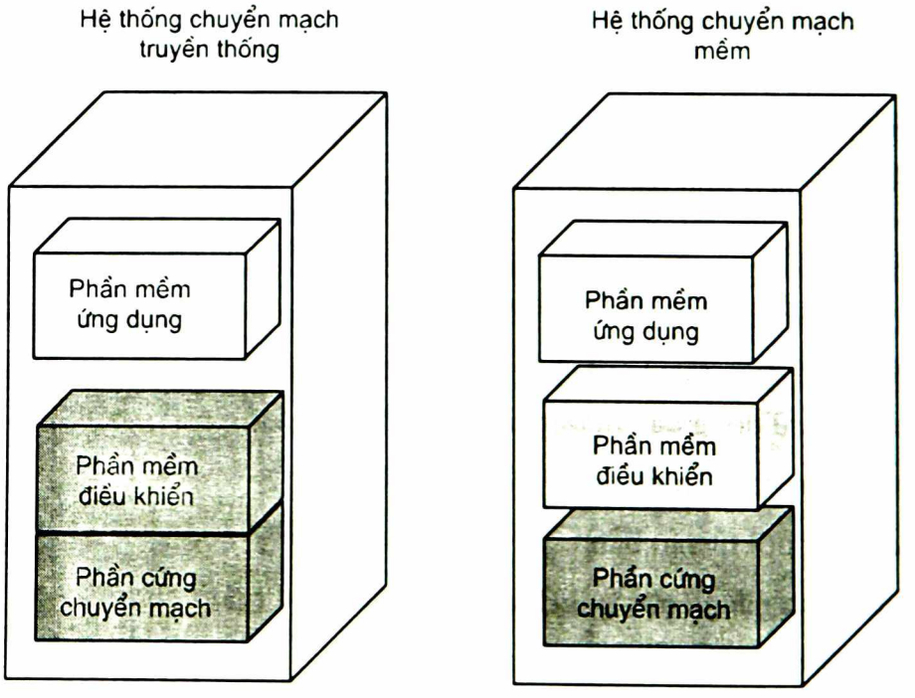
Từ trình bày trên đây về mô hình và các thực thể chức năng của hệ thống chuyển mạch mềm chúng ta có thể khái quát công nghệ chuyển mạch mềm như sau: Chuyển

mạch mềm là hệ thống phần mềm đảm bảo khả năng báo hiệu và điều khiển cuộc gọi trong mạng thế hệ mới NGN. Chuyển mạch mềm chuyển lô-gíc điều khiển dịch vụ ra khỏi thiết bị chuyển mạch tới cơ sở dữ liệu mạng hay các máy chủ ứng dụng AS (trong mạng PSTN 80% dịch vụ do các tổng đài chuyển mạch kênh truyền thống cung cấp). Ngày nay, chuyển mạch mềm thường chạy trên hệ điều hành "***SUN Solaris***".

3.3 SO SÁNH CHUYỂN MẠCH MỀM VÀ CHUYỂN MẠCH TRUYỀN THỐNG

3.3.1 Kiến trúc các hệ thống chuyển mạch

Chuyển mạch mềm thực hiện các chức năng tương tự chuyển mạch kênh truyền thống nhưng với năng lực mềm dẻo và các tính năng ưu việt hơn. Các ưu điểm của chuyển mạch mềm mang lại là do ở hệ thống chuyển mạch mềm chức năng chuyển mạch được thực hiện bằng phần mềm dựa trên cấu trúc phân tán và các giao diện lập trình ứng dụng mở. Trên hình 3.3 minh họa mô hình hệ thống chuyển mạch truyền thống và hệ thống chuyển mạch mềm. Từ hình vẽ ta thấy hai hệ thống chuyển mạch này đều gồm 3 khối: phần cứng, phần mềm điều khiển cuộc gọi và phần mềm các dịch vụ ứng dụng.

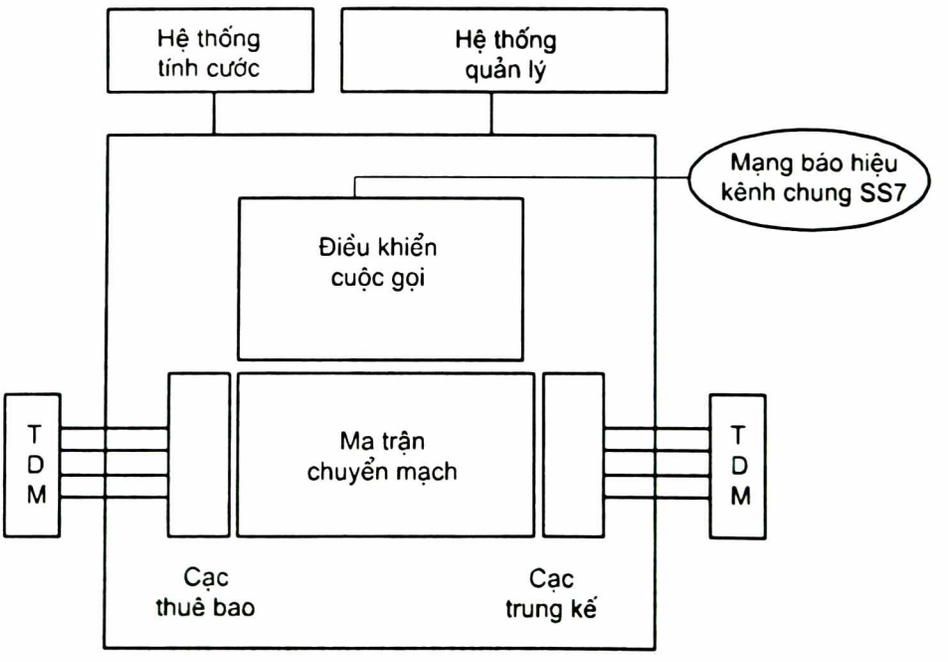


Hình 3.3: Mô hình các hệ thống chuyển mạch

Tuy nhiên trong chuyển mạch truyền thống tức hệ thống chuyển mạch điện thoại sử dụng trong mạng PSTN, phần cứng chuyển mạch luôn đi kèm với phần mềm điều

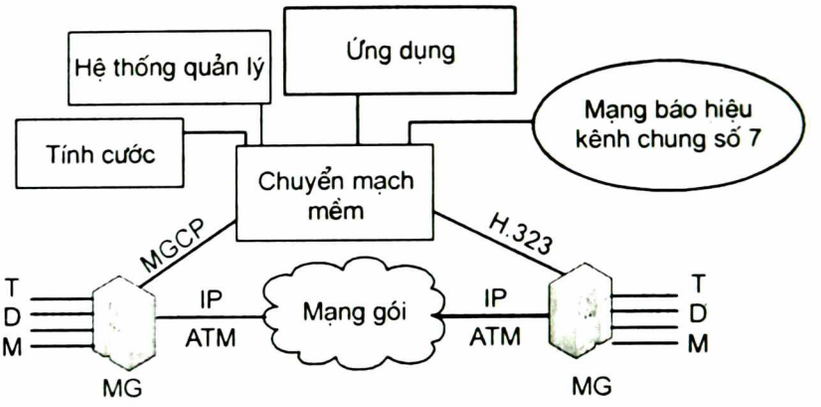
khiển cuộc gọi và phần mềm ứng dụng của cùng một nhà cung cấp. Điều này làm tăng tính độc quyền trong việc cung cấp các hệ thống chuyển mạch, không đảm bảo một môi trường tạo dịch vụ mới do đó làm giới hạn khả năng phát triển các dịch vụ mới của các nhà quản trị mạng. Khắc phục điều này, chuyển mạch mềm đưa ra giao diện lập trình ứng dụng mở, có khả năng tương thích phần mềm điều khiển và phần cứng của các nhà cung cấp khác nhau. Điều này cho phép các nhà cung cấp phần mềm và phần cứng có được quan điểm chung và tập trung vào lĩnh vực của mình. Với các giao diện lập trình mở, chuyển mạch mềm có thể dễ dàng nâng cấp, thay thế và tương thích với ứng dụng của các nhà cung cấp khác nhau. Chuyển mạch mềm được xây dựng trên cơ sở công nghệ mạng IP, xử lý thông tin một cách trong suốt, cho phép đáp ứng nhiều loại lưu lượng khác nhau. Được xây dựng theo cấu hình phân tán, tách các chức năng điều khiển cuộc gọi và ứng dụng của hệ thống ra khỏi chức năng chuyển mạch làm cho nhiệm vụ chuyển mạch trở nên đơn giản hơn và do đó năng lực xử lý mạnh mẽ hơn, linh hoạt và hiệu quả hơn. Đồng thời công nghệ chuyển mạch mềm làm giảm tính độc quyền của các nhà cung cấp, góp phần tăng tính cạnh tranh và do đó giảm giá thành của hệ thống chuyển mạch.

Các phân tích trên đây được làm sáng tỏ thêm khi nghiên cứu các hình 3.4 và hình 3.5 minh họa sơ đồ khối chức năng của tổng đài chuyển mạch kênh truyền thống và tổng đài chuyển mạch mềm.



Hình 3.4: Sơ đồ chức năng hệ thống chuyển mạch kênh

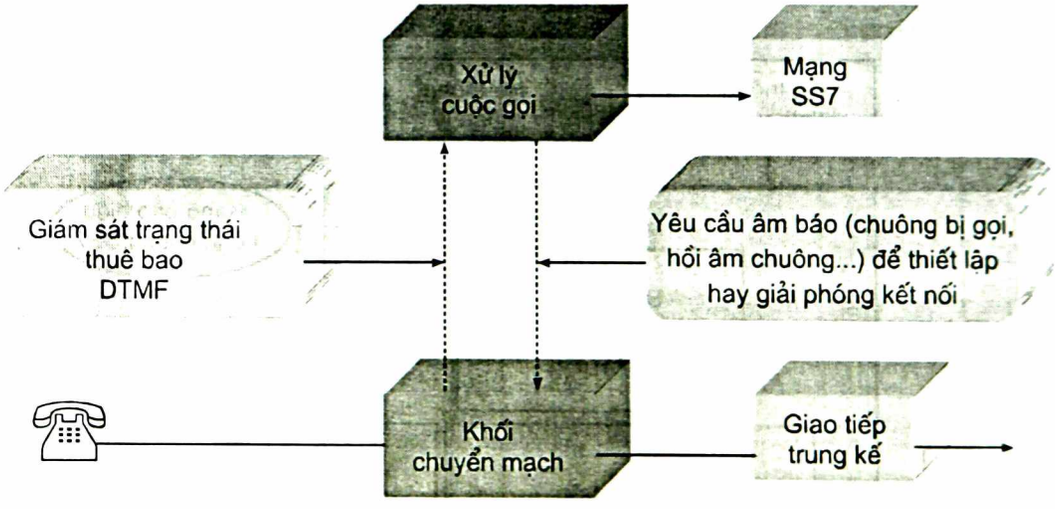
Hình 3.5 dưới đây minh họa sơ đồ khối chức năng của chuyển mạch mềm.



Hình 3.5: Sơ đồ chức năng hệ thống chuyển mạch mềm

3.3.2 Phương thức xử lý cuộc gọi

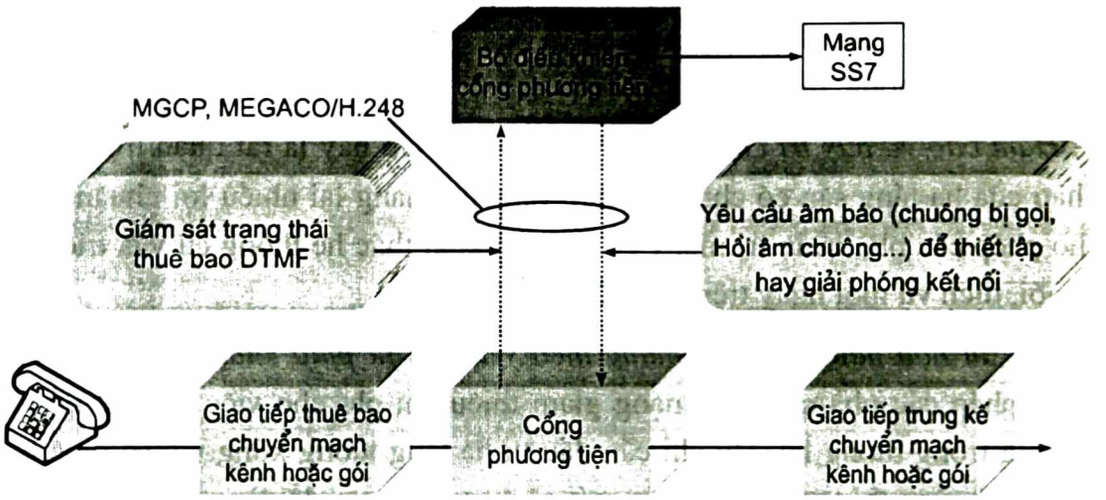
Tổng đài truyền thống xây dựng theo cấu trúc tập trung (hình 3.5), chức năng xử lý cuộc gọi và chức năng chuyển mạch vật lý luồng lưu lượng đều được tập trung trong một tổng đài. Thiết bị này đồng thời vừa giám sát thuê bao, đưa ra các xử lý theo tiến trình cuộc gọi vừa thực hiện chức năng chuyển mạch các luồng lưu lượng.



Hình 3.6: Hoạt động chuyển mạch kênh truyền thống

Trong khi đó hệ thống chuyển mạch mềm (hình 3.6) với cấu trúc phân tán, chức năng xử lý cuộc gọi (Media Gateway Controller) và chuyển mạch vật lý các luồng lưu lượng (Media Gateway) được tách rời nhau. Hai chức năng này giao tiếp với nhau qua các giao diện lập trình ứng dụng với các giao thức báo hiệu và điều khiển như MGCP,

MEGACO/H.248. Trong hệ thống chuyển mạch mềm các Media Gateway MG thực hiện chức năng giám sát trạng thái các đầu cuối, sau đó gửi thông tin về cho Media Gateway Controller MGC xử lý, MGC xử lý cuộc gọi và điều khiển MG chuyển mạch vật lý để tạo kênh truyền thông lưu lượng. Các bản tin báo hiệu số 7 được thu và xử lý bởi cổng báo hiệu SG.



Hình 3.7: Hoạt động của chuyển mạch mềm

Tóm lại, công nghệ chuyển mạch truyền thống tách biệt hai phương thức chuyển mạch kênh và chuyển mạch gói, trong khi đó chuyển mạch mềm kết hợp và hoàn thiện chúng trong mạng hội tụ và tích hợp, do đó hệ thống chuyển mạch mềm đảm bảo tính đa năng, kinh tế, mềm dẻo và hiệu quả cao hơn so với hệ thống chuyển mạch truyền thống.

3.3.3 Các ưu điểm của chuyển mạch mềm

Quan điểm của các nhà quản trị mạng

Nhìn từ phía nhà quản trị mạng, có rất nhiều lợi ích để họ tăng tính cạnh tranh và giảm chi phí đầu tư CAPEX cũng như chi phí điều hành mạng OPEX (tất cả sẽ đem lại lợi ích cho khách hàng):

- *Môi trường kiến tạo dịch vụ mới*: Với giao diện ứng dụng lập trình mở các nhà quản trị mạng có thể tự mình đưa ra các ứng dụng và dịch vụ tới các khách hàng của mình. Điều này khiến họ hoàn toàn chủ động trong việc đưa ra các chiến lược kinh doanh và phát triển dịch vụ một cách lâu dài, ít phụ thuộc vào các nhà sản xuất.

- *Khả năng vận hành và bảo dưỡng từ xa*: Các nhà quản trị mạng thực hiện vận hành và bảo dưỡng các thiết bị phân bố của mình thông qua mạng với các giao diện trực quan. Hệ thống này cho phép họ có thể truy cập tới từng tài nguyên thuộc từng cạc riêng biệt thông qua một máy chủ đặt ở bất kỳ nơi nào.

- *Giảm thời gian triển khai dịch vụ và hệ thống*: Vì có một kiến trúc mở do đó việc triển khai hệ thống có thể thực hiện bằng việc nâng cấp các thiết bị hiện có và lắp đặt thêm các MG, MGC, SG, AS. Dịch vụ đưa ra được triển khai bằng phần mềm trên các mô-đun API. Tất cả điều đó làm giảm thời gian tiếp cận thị trường của nhà quản trị, kể từ khi bỏ vốn đầu tư. Một đặc điểm rất quan trọng đó là nhờ việc ứng dụng các giao diện lập trình ứng dụng mở API nhà cung cấp dịch vụ không còn hoàn toàn bị phụ thuộc vào nhà sản xuất, do đó việc tạo ra và đưa dịch vụ mà thị trường có thể thực hiện một cách nhanh chóng, tiện ích và hiệu quả cao.

- *Đảm bảo và duy trì cơ sở hạ tầng sẵn có*: Điều này là rất quan trọng trong khi khấu hao của hạ tầng sẵn có chưa hết hay vẫn còn mang lại nhiều lợi nhuận. Ưu điểm này cho phép các nhà quản trị mạng có thể vừa khai thác hệ thống cũ vừa triển khai hệ thống mới, dịch vụ mới ngay trên nền hệ thống cũ.

- *Tối ưu hóa năng lực của mạng theo yêu cầu từ phía khách hàng*: Đây là khả năng cho phép các nhà quản trị mạng giảm thiểu chi phí cho việc dự phòng gia tăng khách hàng trong tương lai. Họ chỉ cần đầu tư hạ tầng tương xứng với số thuê bao của mình, khi mở rộng thị trường, các cạc mới sẽ được bổ sung dễ dàng. Ưu điểm này cũng tạo điều kiện kinh doanh cho những người có vốn hạn chế, với số ít khách hàng và đó cũng chính là khả năng chống độc quyền, khuyến khích phát triển đồng thời tạo điều kiện cho nhiều đối tác cùng khai thác, đầu tư và cạnh tranh.

- Khả năng nâng cấp hệ thống cao, thúc đẩy việc ứng dụng các công nghệ mới: Với kiến trúc mở, hạ tầng mạng có thể dễ dàng được nâng cấp và việc ứng dụng công nghệ mới không làm ảnh hưởng tới hạ tầng mạng sẵn có.

Cuối cùng ta có thể đưa ra một bảng so sánh các thuộc tính của chuyển mạch mềm và chuyển mạch kênh truyền thống.

Bảng 3.1: So sánh các thuộc tính của chuyển mạch mềm và chuyển mạch kênh truyền thống

	Chuyển mạch mềm	**Chuyển mạch trong tổng đài PSTN truyền thống**
Phương pháp chuyển mạch	Trên cơ sở phần mềm	Trên cơ sở các cấu kiện điện tử
Kiến trúc	Phân tán, theo các chuẩn mở	Riêng biệt của từng nhà sản xuất và tập trung cao
Khả năng thích hợp với ứng dụng của nhà cung cấp khác	Dễ dàng (API mở)	Khó khăn
Khả năng thay đổi mềm dẻo, linh hoạt	Có	Khó khăn

	Chuyển mạch mềm	Chuyển mạch trong tổng đài PSTN truyền thống
Giá thành	Rẻ hơn, khoảng bằng một nửa tổng đài điện tử truyền thống	Đắt hơn
Khả năng nâng cấp	Rất cao và nhanh	Tốt, tuy có hạn chế hơn và chậm
Giá thành của cấu hình cơ bản	Thấp, giá thành thay đổi gần như tuyến tính theo số lượng thuê bao. Cấu hình cơ bản có thể sử dụng cho mạng doanh nghiệp	Rất cao, tổng đài PSTN không thích hợp cho mạng doanh nghiệp.
Khả năng kết nối đồng thời	Vài trăm đến vài triệu kết nối đồng thời	Vài trăm nghìn đến 1 triệu kết nối đồng thời
Dung lượng ban đầu nhỏ	Có thể xây dựng tổng đài dung lượng vài trăm số	Dung lượng ban đầu yêu cầu rất lớn
Truyền thông đa phương tiện	Có	Rất hạn chế
Hội nghị truyền hình	Tốt hơn	Có, nhưng hạn chế
Lưu lượng	Thoại, fax, dữ liệu, video...	Chủ yếu là thoại và fax
Thiết kế cho độ dài cuộc gọi	Không hạn chế	Ngắn (chỉ vài phút)
Tạo dịch vụ mới	Dễ, nhanh và tự làm	Khó, chậm và phụ thuộc vào nhà sản xuất

Quan điểm của khách hàng

Nhìn từ phương diện khách hàng, họ được lợi về mặt giá cả và sự đa dạng về dịch vụ khi triển khai chuyển mạch mềm, cụ thể là:

- Nâng cao tiện ích sử dụng nhờ sự phát triển công nghệ và khả năng hội tụ công nghệ và tích hợp dịch vụ.
- Giảm giá thành nhờ tính cạnh tranh của thị trường.
- Đáp ứng mọi yêu cầu và nhu cầu của người dùng

Bảng 3.2 tổng kết những ưu việt nổi trội của công nghệ chuyển mạch mềm.

Bảng 3.2: Những ưu việt nổi trội của công nghệ chuyển mạch mềm

Giảm giá thành	Cải thiện dịch vụ	Chuyển sang mạng IP
Quản lý mạng hiệu quả hơn	Dịch vụ phân biệt và gia tăng giá trị	Tận dụng các cơ sở dữ liệu mạng hiện có
Giảm giá xử lý cuộc gọi	Đưa dịch vụ mới ra ứng dụng nhanh hơn	Tận dụng mạng và dịch vụ nhờ kiến trúc phân tán
Tăng dung lượng nhờ giảm tải Internet	Cung cấp khả năng cho người dùng tự quản lý dịch vụ của mình	Dễ mở rộng

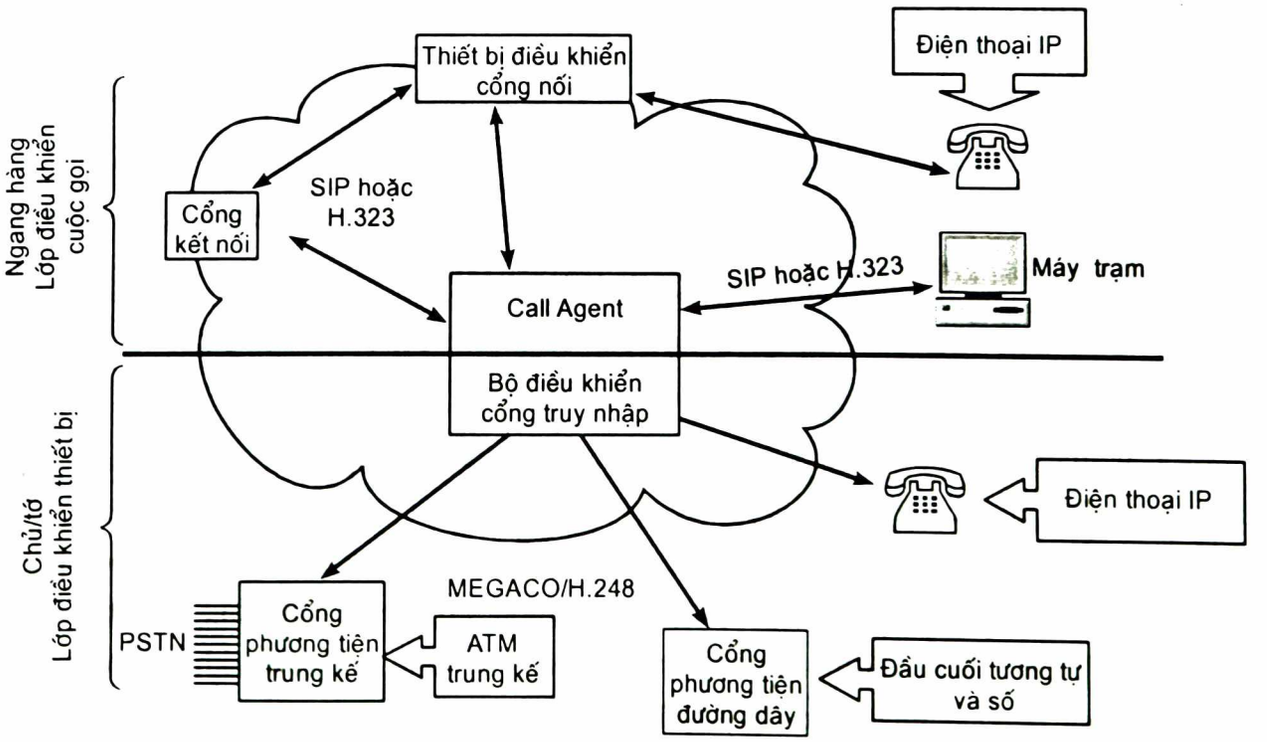
Chương 4

CÁC GIAO THỨC CƠ BẢN CỦA CHUYỂN MẠCH MỀM

4.1 TỔNG QUAN VỀ BÁO HIỆU TRONG CHUYỂN MẠCH MỀM

Hệ thống chuyển mạch mềm có kiến trúc phân tán, các chức năng báo hiệu và xử lý báo hiệu, chuyển mạch, điều khiển cuộc gọi... được thực hiện bởi các thiết bị phân tán trong cấu hình mạng. Để có thể tạo ra các kết nối giữa các đầu cuối nhằm cung cấp dịch vụ cho người dùng các thiết bị này phải trao đổi các thông tin báo hiệu. Cách thức trao đổi các thông tin báo hiệu được quy định bởi các giao thức báo hiệu. Các giao thức báo hiệu cơ bản trong hệ thống chuyển mạch mềm bao gồm:

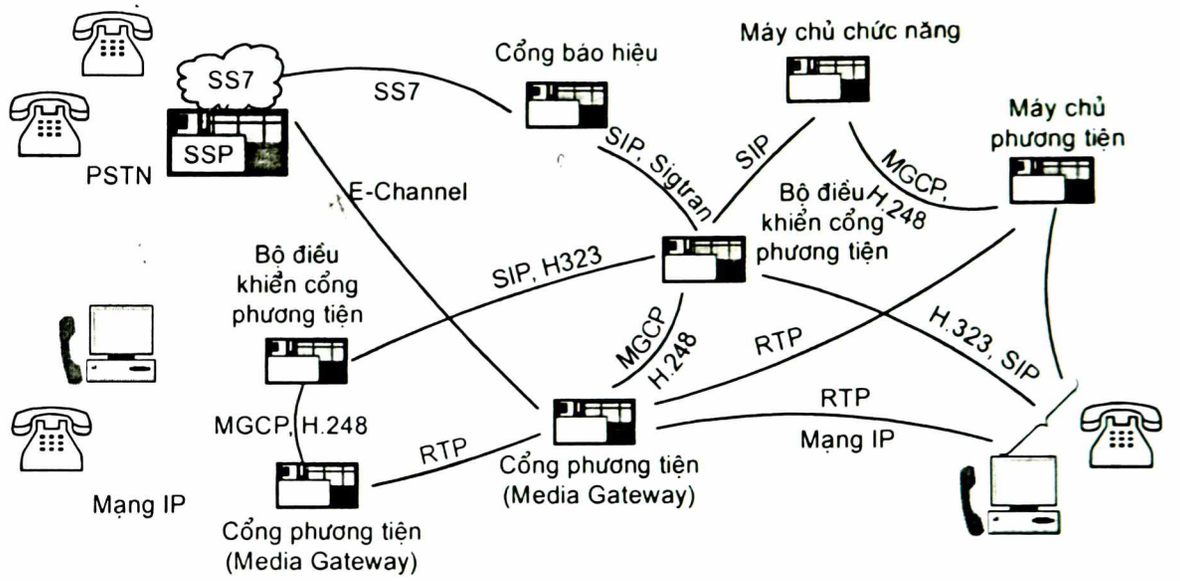
- H.323
- SIP
- MGCP, MEGACO/H.248
- SIGTRAN



Hình 4.1: Phân loại giao thức báo hiệu trong chuyển mạch mềm

Các giao thức này có thể phân thành hai loại: giao thức ngang cấp (H.323, SIP) và giao thức Chủ-tớ (MGCP, MEGACO/H.248). Mỗi loại giao thức có ưu điểm và nhược điểm riêng của mình với các chức năng khác nhau, tồn tại trong mạng ở các cấp khác nhau. Hình 4.1 minh họa vị trí và phạm vi sử dụng các giao thức trong mạng.

Vị trí và vai trò của từng giao thức trên có thể được minh họa trên hình 4.2.



Hình 4.2: Các giao thức báo hiệu trong mạng chuyển mạch mềm

• Giao thức ngang cấp H.323, SIP được sử dụng để trao đổi thông tin báo hiệu và điều khiển giữa các MGC, giữa MGC và các máy chủ.

• Giao thức chủ tớ MGCP, MEGACO/H.248 là giao thức báo hiệu điều khiển giữa MGC và các cổng kết nối (trong đó MGC điều khiển cổng kết nối).

• Giao thức SIGTRAN là giao thức báo hiệu giữa MGC và cổng báo hiệu. Chức năng của giao thức này là chuyển đổi báo hiệu SS7 và IP.

Các giao thức ngang cấp thực hiện chức năng mạng ở cấp cao hơn, quy định cách thức giao tiếp giữa các thực thể cùng cấp để cùng phối hợp thực hiện cuộc gọi hay các ứng dụng khác. Trong khi đó các giao thức chủ-tớ là sản phẩm của việc phân bố không đồng đều trí tuệ mạng, phần lớn trí tuệ mạng được tập trung trong các thực thể chức năng điều khiển (đóng vai trò là chủ), thực thể này sẽ giao tiếp (điều khiển) với nhiều thực thể khác qua các giao thức chủ-tớ nhằm cung cấp dịch vụ.

Bảng 4.1 so sánh các thuộc tính cơ bản của hai loại giao thức mạng thường dùng.

Bảng 4.1: Bảng so sánh các giao thức báo hiệu và điều khiển

Loại giao thức	Chủ-tớ	Ngang cấp
Vận hành-Khai thác	- Các thiết bị/Cổng đơn giản/ít trí tuệ - Trí tuệ ứng dụng chứa trong máy chủ mạng	- Các thiết bị/cổng phức tạp/thông minh - Tương tác ngang cấp
Phát triển dịch vụ	- Phát triển trên cơ sở máy chủ. Công cụ chung. - Nhanh chóng hơn đưa dịch vụ mới ra thị trường thông qua các mạng khác nhau.	- Các thiết bị chuyên dụng - Có thể
Phát triển dịch vụ (đối với nhà khai thác)	- Chỉ cập nhật các dịch vụ điều khiển - Các dịch vụ có thể quản lý được một cách động qua mạng	- Cần phải nâng cấp tất cả các thiết bị cổng để triển khai các tính năng dịch vụ mới qua mạng
Giá thành	- Tối ưu hoá giá thành các thiết bị cổng do vậy cho phép giảm giá thành của toàn hệ thống (đặc biệt là đối với các hệ thống lớn). - Thời hạn của các thiết bị cổng dài hơn	- Giá thành của các cổng đơn khối cao - Giá thành hệ thống cao (đặc biệt là đối với các hệ thống lớn). - Các thiết bị cổng có thể cần phải nâng cấp phần cứng thường xuyên
Ví dụ	- MEGACO/H.248 - MGCP - NCS/J.162 (Phiên bản cũ của MGCP)	- SIP - H.323

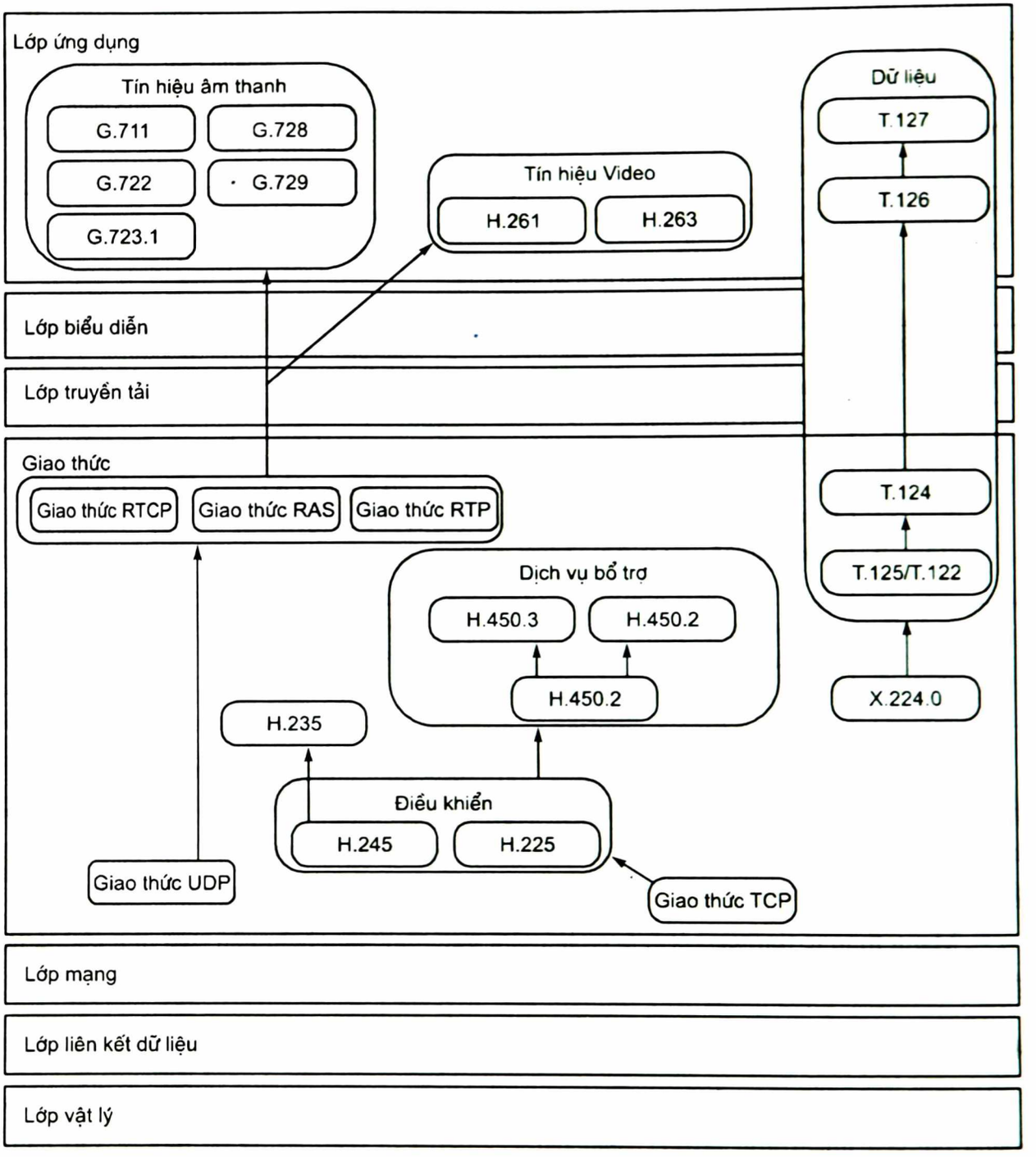
Từ bảng thống kê trên đây chúng ta thấy rõ là giao thức chủ-tớ cho phép tập trung phần trí tuệ ứng dụng vào một số ít các máy chủ điều khiển, do đó có thể dễ nâng cao hiệu quả của các thiết bị cổng kết nối và có thể nhanh hơn trong việc đưa dịch vụ ứng dụng mới vào thị trường vì chỉ cần cặp nhật mới cho phần tập trung như các máy chủ.

Mạng thực tế thường kết hợp cả hai loại giao thức báo hiệu nêu trên như chúng ta đã thấy rõ trên các hình 4.1 và hình 4.2.

4.2 GIAO THỨC H.323

Giao thức H.323 là tiêu chuẩn dành cho truyền thông đa phương tiện trên cơ sở mạng chuyển mạch gói, do ITU ban hành. Phiên bản đầu tiên được đưa ra vào năm 1996 và phiên bản 4 là phiên bản đang được sử dụng rộng rãi được ban hành vào năm 2001. Phiên bản 1 và 2 hỗ trợ H.245 trên nền TCP, Q.931 trên nền TCP và RAS trên nền UDP. Các phiên bản 3 và 4 có hỗ trợ thêm H245 và Q.931 trên nền TCP và UDP. Ban đầu H.323 dự định dành cho X.25 và ATM nhưng trong thực tế nó lại được biết đến nhiều hơn với ứng dụng VoIP.

Ngăn xếp giao thức H.323 theo mô hình OSI:



Hình 4.3: Ngăn xếp giao thức H.323.

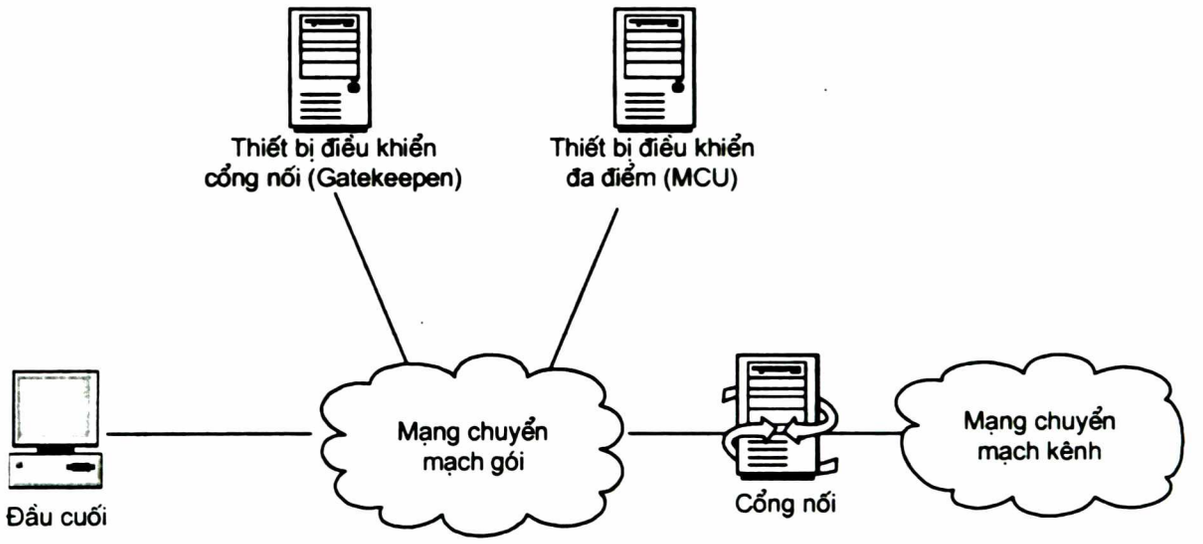
Từ ngăn xếp giao thức H.323 trình bày trên đây ta thấy:

• H.323 hỗ trợ cho lưu lượng thoại qua các chuẩn mã hoá G.711, G.722, G.728, G.729, G.723.1 trên nền giao thức RTP và RTCP thông qua phương thức chuyển tải không tin cậy UDP.

• H.323 hỗ trợ lưu lượng video qua các chuẩn mã hoá H.261, H.263 trên nền các giao thức RTP và RTCP cũng qua phương thức truyền tải không tin cậy (UDP).

• H.323 hỗ trợ lưu lượng dữ liệu các giao thức T.122, T.124, T.125, T.127, T.126 qua phương thức truyền tải tin cậy (TCP).

• H.245 là giao thức điều khiển kênh (trao đổi các thuộc tính cuộc gọi giữa 2 đầu cuối). H.225 là giao thức báo hiệu giữa đầu cuối và gatekeeper (thiết bị điều khiển cổng kết nối).



Hình 4.4: Cấu hình mạng H.323

Cấu hình mạng H.323 (hình 4.4) bao gồm các thành phần sau:

• Đầu cuối: Thiết bị đầu cuối H.323 bắt buộc phải hỗ trợ những giao thức:

- Báo hiệu điều khiển cuộc gọi H.225
- Báo hiệu điều khiển kênh H.245
- Giao thức RTP/RTCP cho dữ liệu
- Các loại bộ mã hóa thoại

Việc hỗ trợ các mã hóa video là không bắt buộc đối với các đầu cuối H.323.

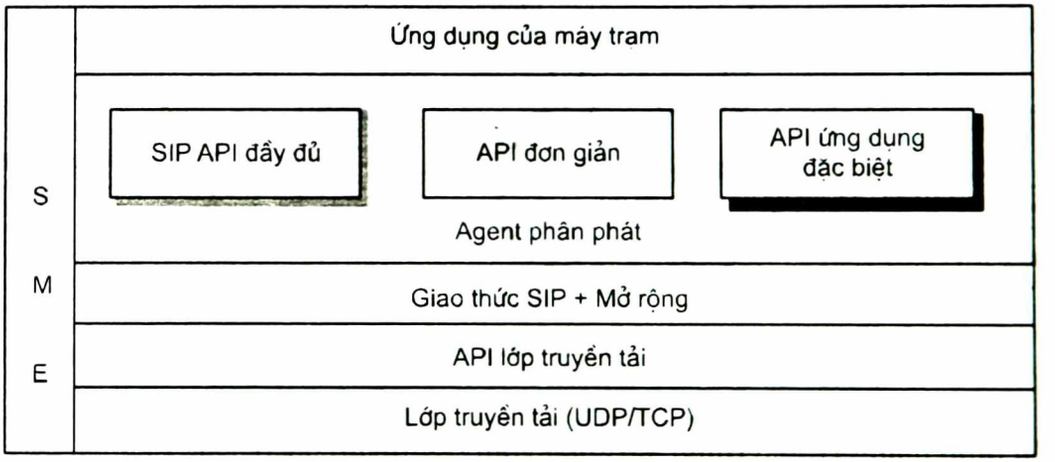
• Cổng nối đảm nhiệm chức năng chuyển đổi giữa hai mạng, thí dụ giữa mạng chuyển mạch gói và mạng PSTN.

• Gatekeeper có chức năng chính là chuyển đổi địa chỉ và điều khiển băng thông. Trong mạng H.323 không nhất thiết phải có Gatekeeper, tuy nhiên nếu có Gatekeeper thì tất cả các đầu cuối phải đăng ký trước khi thực hiện cuộc gọi.

• H.323 MCU (Multipoint Control Unit - Đơn vị điều khiển đa điểm) hỗ trợ các cuộc gọi hội nghị của 3 hay nhiều hơn các đầu cuối cùng tham gia trong một phiên liên lạc. Trong MCU có hai mô-đun: Bộ điều khiển đa điểm MC (Multipoint Controller) có chức năng điều khiển và bộ xử lý đa điểm MP (Multipoint Processor) nhận và xử lý các luồng dữ liệu thoại, video hoặc dữ liệu khác.

4.3 GIAO THỨC SIP

Giao thức SIP do nhóm làm việc MMUSIC (Multiparty Multimedia Session Control) của IETF phát triển từ tiêu chuẩn RFC 2543. Đây là giao thức báo hiệu lớp ứng dụng có chức năng mô tả việc khởi tạo, thay đổi và hủy phiên truyền thông đa phương tiện giữa các đầu cuối. SIP được đưa ra trên cơ sở nguyên lý giao thức trao đổi thông tin của mạng Internet (HTTP). SIP là giao thức ngang cấp, hoạt động theo nguyên tắc hỏi/đáp (server/client). Vị trí giao thức SIP trong mô hình phân lớp hệ thống được mô tả như hình 4.5.



Hình 4.5 Giao thức SIP trong mô hình phân lớp hệ thống

Từ hình 4.5 trên đây chúng ta thấy rằng giao thức SIP có thể chạy trên nền cả UDP và TCP với IPv4 hoặc IPv6. SIP được thiết kế độc lập với các giao thức truyền dẫn mức thấp và có thể bổ sung các tính năng mới thông qua việc thay đổi các tham số hay tiêu đề của các bản tin.

Các thực thể mạng của giao thức SIP bao gồm:

• Đầu cuối SIP: Có thể là máy điện thoại SIP hay máy tính chạy phần mềm SIP. Mỗi đầu cuối sẽ được gán một địa chỉ SIP URL để định danh và nhận thực.

• Proxy Server (máy chủ ủy quyền) là đại diện cho một nhóm các đầu cuối SIP, có nhiệm vụ đáp ứng các yêu cầu SIP của nhóm đó hay từ các proxy khác. Trong trường hợp nó không đáp ứng được thì yêu cầu sẽ được chuyển cho một proxy khác.

• Redirect Server (máy chủ chuyển đổi địa chỉ): Nhận địa chỉ SIP và gửi lại cho nơi hỏi dưới dạng địa chỉ khác của proxy kế tiếp để liên lạc.

• Registrar Server (máy chủ đăng kí): Có chức năng nhận thực, bảo mật. Ghi lại địa chỉ SIP và địa chỉ IP của đầu cuối SIP đăng ký. Nói cách khác là máy chủ này thực hiện các chức năng đăng ký và quản lý các đầu cuối (Endpoints) và địa chỉ của chúng.

• Location Server (máy chủ định vị) là phần mềm định vị đầu cuối, cung cấp thông tin về những vị trí có thể của phía bị gọi cho các phần mềm Proxy Server và Redirect Server.

SIP sử dụng phương pháp mã hoá kiểu văn bản với cấu trúc bản tin theo kiểu hỏi/đáp, bao gồm các bản tin sau:

• Bản tin INVITE - Bắt đầu thiết lập cuộc gọi bằng cách gửi bản tin mời đầu cuối khác tham gia

• Bản tin ACK - Bản tin này khẳng định client (máy trạm) đã nhận được bản tin trả lời bản tin INVITE

• Bản tin BYE - Bắt đầu kết thúc cuộc gọi

• Bản tin CANCEL - Hủy yêu cầu đang nằm trong hàng đợi

• Bản tin REGISTER - Đầu cuối SIP sử dụng bản tin này để đăng ký với Registrar Server

• Bản tin OPTIONS - Sử dụng để xác định năng lực của máy chủ

• Bản tin INFO - Sử dụng để tải các thông tin như âm báo DTMF

Trong phiên hội thoại SIP, mỗi bên tham gia được gán một địa chỉ SIP, các user (người sử dụng) sẽ đăng ký vị trí đăng nhập với máy chủ SIP. Quá trình thiết lập một phiên liên lạc SIP diễn ra như sau:

1. Đăng kí, khởi tạo và định vị đầu cuối

2. Mô tả phiên của đầu cuối được mời tham gia (âm thanh, video hay đa phương tiện)

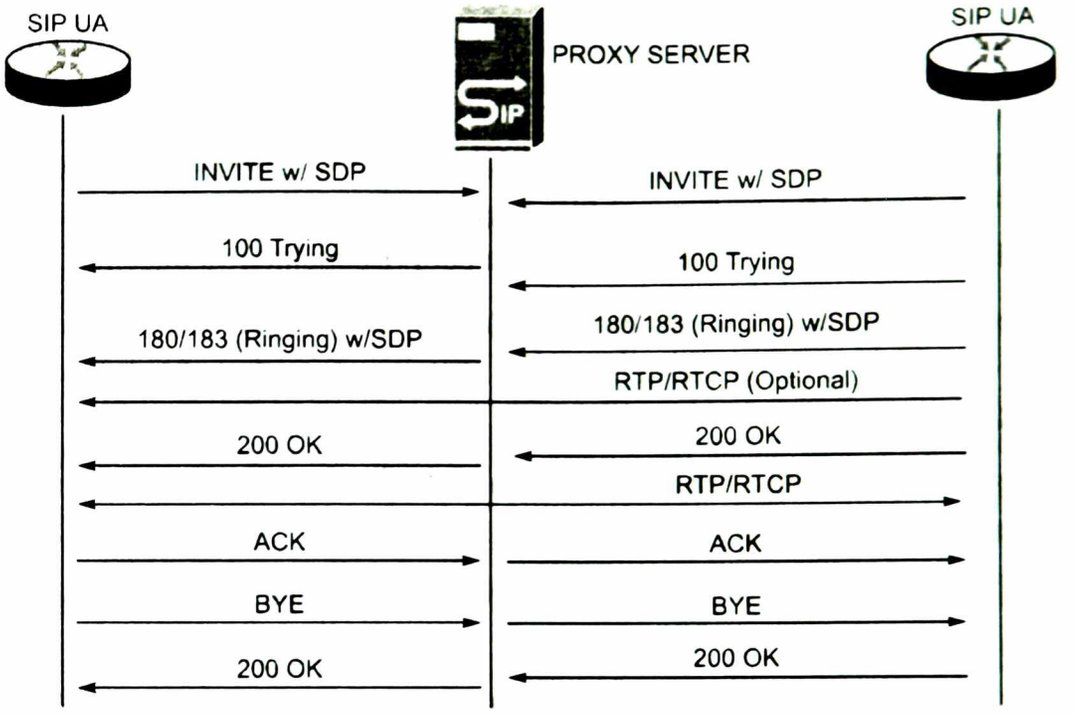
3. Xác nhận thái độ hưởng ứng của đầu cuối được mời

4. Nếu đầu cuối được mời có thái độ tích cực, cuộc gọi được thiết lập

5. Các nhiệm vụ xử lý trong quá trình cuộc gọi

6. Giải phóng cuộc gọi

Chi tiết các quá trình xử lý thiết lập của phiên liên lạc SIP nêu trên được minh họa trên hình 4.6.



Lưu ý: Báo hiệu SIP có thể chạy trên UDP hay TCP

Hình 4.6 Thiết lập và hủy cuộc gọi SIP

Các bản tin trả lời các bản tin SIP nêu trên gồm có:

1xx - Các bản tin chung

2xx - Thành công

3xx - Chuyển địa chỉ

4xx - Yêu cầu không được đáp ứng

5xx - Sự cố của máy chủ

6xx - Sự cố hòa mạng

Chi tiết cụ thể về giao thức báo hiệu SIP sẽ được trình bày trong chương 5.

4.4 GIAO THỨC MGCP-MEGACO/H.248

Một trong những giao thức quan trọng nhất, là nền tảng của chuyển mạch mềm là giao thức điều khiển cổng phương tiện MGCP (Media Gateway Control Protocol). Giao thức này quy định cách thức mà MGC điều khiển các MG trong việc thiết lập kết nối khi mà các phần mềm điều khiển, xử lý cuộc gọi được tách khỏi các MG. MGCP được phát triển từ hai giao thức ban đầu là: giao thức điều khiển cổng đơn giản SGCP (Simple Gateway Control Protocol) và giao thức điều khiển thiết bị Internet IPDC (Internet Protocol Device Control). MGCP là giao thức được thiết kế dành chủ yếu để xử lý báo hiệu trong mạng IP, giao thức này ít có năng lực xử lý cho việc truyền tải các gói thoại

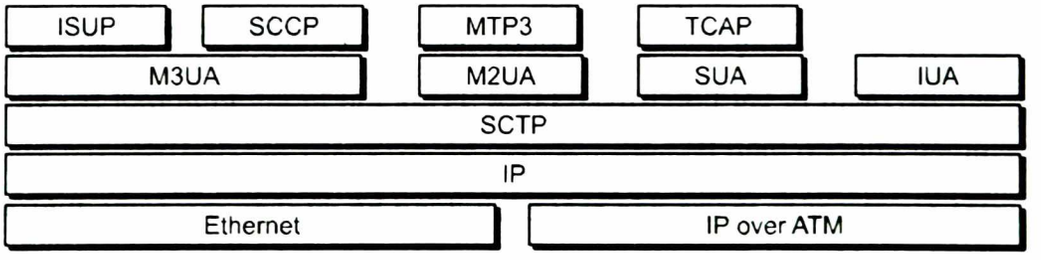
nói chung, ví dụ như VoATM. Việc ứng dụng MGCP trong thực tế cũng chỉ ra một số nhược điểm của giao thức này như thiếu một phương thức hữu hiệu để MGC có thể thu thập thông tin về khả năng xử lý của một MG nào đó. Những nhược điểm này được khắc phục bởi một giao thức khác với các điểm kế thừa từ MGCP và các tính năng tăng cường, giao thức MEGACO (MEdia GAteway COtrol protocol) do IETF phát triển. Tổ chức ITU đã phối hợp với IETF để phát triển giao thức tương tự như MEGACO và gọi tên là giao thức H.248.

Chương 6 tiếp theo sẽ xem xét chi tiết về hai giao thức này.

4.5 GIAO THỨC SIGTRAN

Mạng NGN là sự liên kết mạng và hội tụ của công nghệ thông tin, mạng IP, và internet vào mạng viễn thông. Sự hội tụ này đặt ra yêu cầu chuyển tiếp giữa mạng PSTN truyền thống và mạng IP. Điều này được tiến hành nhờ giao thức truyền tải báo hiệu SIGTRAN (SIGnaling TRANsport).

Hình 4.7 trình bày ngăn xếp giao thức SIGTRAN



Hình 4.7: Ngăn xếp giao thức SIGTRAN

SIGTRAN là một giao thức hỗ trợ cho mạng điện thoại IP (VoIP), nó cho phép truyền tải các bản tin báo hiệu số 7 qua mạng IP, nhờ đó lưu lượng của mạng PSTN và ISDN dựa trên báo hiệu số 7 có thể truyền qua mạng IP. Để làm được điều này, SIGTRAN sử dụng một loạt các giao thức thành phần và các mô-đun tương thích bao gồm: Giao thức truyền tải điều khiển dòng SCTP (Stream Control Transport Protocol), mô-đun tương thích với MTP lớp 3 (M3UA), mô-đun tương thích với MTP lớp 2 (M2UA), lớp tương thích với người dùng ISDN (IUA).

- SCTP có các chức năng:

  - Truyền tải các bản tin báo hiệu của mạng PSTN qua mạng IP

- Là giao thức truyền tải tin cậy trên nền mạng gói theo chế độ không hướng kết nối.
- Xác nhận không có lỗi và không truyền lại dữ liệu.
- Phân mảnh dữ liệu (cắt một bản tin khách hàng của SCTP và đưa vào nhiều gói)
- Đóng gói dữ liệu (gộp nhiều bản tin là khách hàng vào một gói)

• M3UA có các chức năng:

- Hỗ trợ chuyển giao các bản tin MTP lớp 3 (ISUP, SCCP, TUP...)
- Hỗ trợ cho hoạt động của các giao thức lớp MTP3
- Hỗ trợ quản lý kết hợp truyền tải SCTP và lưu lượng giữa một SG và một hay nhiều MGC
- Hỗ trợ cho MGC hay các cơ sở dữ liệu lưu trữ chia sẻ lưu lượng

• SUA có các chức năng:

- Hỗ trợ chuyển giao các bản tin phần người dùng SCCP (ví dụ như TCAP, RANAP)
- Hỗ trợ các dịch vụ không hướng kết nối (connectionless oriented) SCCP
- Hỗ trợ các dịch vụ hướng kết nối (connection oriented) SCCP
- Hỗ trợ quản lý kết hợp truyền tải SCTP và lưu lượng giữa một SG và một hay nhiều nút báo hiệu IP
- Hỗ trợ cho việc thông báo một cách đồng bộ sự thay đổi trạng thái cho công tác quản lý.

Chi tiết cụ thể về giao thức SIGTRAN sẽ được trình bày trong chương 7.

Chương 5

GIAO THỨC KHỞI TẠO PHIÊN SIP

5.1 ĐỊNH NGHĨA VÀ CÁC THÀNH PHẦN CỦA HỆ THỐNG SIP

5.1.1 Các định nghĩa và khái niệm cơ bản

Call: Một cuộc gọi gồm tất cả các thành viên trong phiên được mời bởi một tài nguyên chung. Một cuộc gọi SIP được nhận biết bởi một Call-ID (Bộ nhận dạng cuộc gọi) duy nhất.

Call leg: Được nhận biết bởi sự kết hợp của Call-ID, "To" và "From".

Client: Là một chương trình ứng dụng gửi đi những yêu cầu của SIP. Client có thể ảnh hưởng trực tiếp hoặc không đến người sử dụng. Client được chứa trong các Proxy và User Agent.

Conference: Hội nghị là một phiên hội thoại đa phương. Một hội nghị có thể không có hoặc có nhiều thành viên bao gồm các trường hợp như hội nghị đa phương, hội nghị nhiều mắt lưới (full-mesh), cuộc gọi hai thành viên... một vài cuộc gọi có thể tạo ra một hội nghị.

Downstream: là yêu cầu gửi trực tiếp từ phía chủ gọi đến người nghe (từ UAC đến UAS).

Final Response: là các đáp ứng để kết thúc một phiên giao dịch SIP, bao gồm các đáp ứng sau: 2xx, 3xx, 4xx, 5xx, 6xx.

Invitation: Là yêu cầu khởi tạo gửi từ User hoặc Server đề nghị tham gia vào một phiên hội thoại. Một lời mời đầy đủ gồm một yêu cầu INVITE ngay sau một yêu cầu ACK.

Parallel search: Trong một quá trình tìm kiếm song song một Proxy đưa ra một vài yêu cầu tới người dùng hiện tại trong khi nhận một yêu cầu đến.

Provisional Response: Đáp ứng tạm thời là đáp ứng được Server thông báo một tiến trình gọi nhưng chưa kết thúc một phiên giao dịch SIP, đáp ứng 1xx là một đáp ứng Provisional Response.

Server: Là một chương trình ứng dụng có nhiệm vụ nhận các yêu cầu hợp lệ từ các dịch vụ và gửi trả lại các đáp ứng.

Session: Theo đặc tả của SDP thì một phiên đa truyền thông là tập hợp những người gửi và nhận cùng với dòng dữ liệu từ nơi gửi đến nơi nhận. Nó được xác định bởi chuỗi các tên User, Session ID (chỉ số phiên), kiểu mạng, kiểu địa chỉ và địa chỉ các phần tử trong trường nguồn.

Upstream: Đáp ứng gửi trực tiếp từ UAS đến UAC.

URL-encoded: Là chuỗi ký tự mã hóa theo chuẩn RFC 1738.

Đặc tính của các kiểu Server khác nhau được tổng kết trong bảng 5.1:

Bảng 5.1: Đặc tính của các kiểu Server

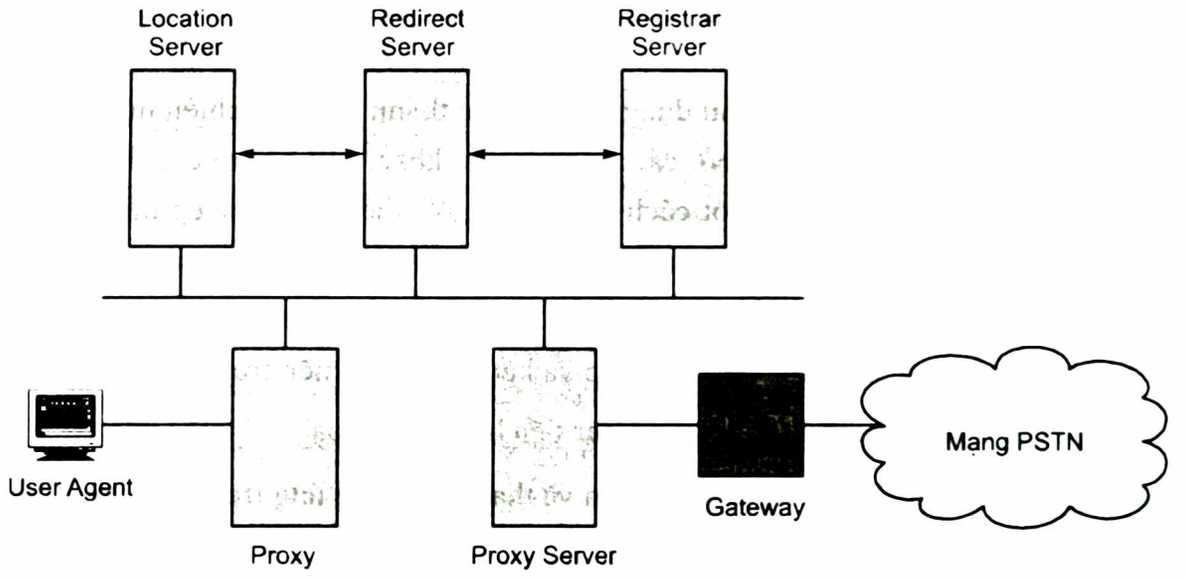
Đặc tính	Redirect Server	Proxy Server	User Agent Server	Registrar Server
Hoạt động như một SIP Client	Không	Có	Không	Không
Trả lại trạng thái 1xx	Có	Có	Có	Có
Trả lại trạng thái 2xx	Không	Có	Có	Có
Trả lại trạng thái 3xx	Có	Có	Có	Có
Trả lại trạng thái 4xx	Có	Có	Có	Có
Trả lại trạng thái 5xx	Có	Có	Có	Có
Trả lại trạng thái 6xx	Không	Có	Có	Có
Chèn tiêu đề (Via)	Không	Có	Không	Không
Chấp nhận (ACK)	Có	Có	Có	Không

5.1.2 Các thành phần của kiến trúc SIP

Xét trên quan điểm Client/Server, các thành phần chính của một hệ thống SIP được mô tả bởi hình 5.1.

Trong hình 5.1, UA (User Agent) là thiết bị đầu cuối trong mạng SIP, nó có thể là một máy điện thoại SIP hay một máy tính chạy phần mềm đầu cuối SIP. UA có thể khởi tạo, thay đổi hay giải phóng cuộc gọi. Trong đó phân biệt hai loại UA: UAC (User Agent Client) và UAS (User Agent Server). UAC là một thực thể thực hiện việc khởi tạo một cuộc gọi còn UAS là một thực thể thực hiện việc nhận cuộc gọi. Nhưng cả UAC và UAS đều có thể giải phóng cuộc gọi.

Proxy Server là phần mềm trung gian hoạt động vừa như Server và vừa như cả Client để thực hiện các yêu cầu thay thế cho các đầu cuối khác. Tất cả các yêu cầu được xử lý tại chỗ bởi Proxy Server (nếu có thể) hoặc nó chuyển đến cho các máy chủ khác. Trong trường hợp Proxy Server không trực tiếp đáp ứng các yêu cầu này thì Proxy Server sẽ thực hiện khâu chuyển đổi hoặc dịch sang khuôn dạng thích hợp trước khi chuyển đi.



Hình 5.1: Cấu trúc của hệ thống SIP

Location Server là phần mềm định vị thuê bao, cung cấp thông tin về những vị trí. Redirect Server là phần mềm nhận yêu cầu SIP và chuyển đổi địa chỉ SIP sang một số địa chỉ khác và gửi lại những địa chỉ này cho đầu cuối. Không giống như Proxy Server, Redirect Server không bao giờ hoạt động như một đầu cuối, tức là không gửi đi bất cứ một yêu cầu nào. Redirect Server cũng không thực hiện việc chấp nhận hay hủy cuộc gọi.

Registrar Server là phần mềm nhận các yêu cầu đăng ký. Trong nhiều trường hợp Registrar Server đảm nhiệm luôn một số chức năng an toàn như xác nhận người sử dụng. Thông thường Registrar Server được cài đặt cùng với Proxy hoặc Redirect Server hoặc cung cấp dịch vụ định vị thuê bao. Mỗi lần đầu cuối được bật lên (thí dụ máy điện thoại hoặc phần mềm SIP) thì đầu cuối lại đăng ký với Server. Nếu đầu cuối cần thông báo với Server về địa điểm của mình thì bản tin Register được gửi đi. Nói chung các đầu cuối đều thực hiện việc đăng ký lại một cách định kỳ.

5.2 CHỨC NĂNG VÀ TÍNH NĂNG CỦA SIP

5.2.1 Các chức năng của SIP

SIP là một giao thức điều khiển tầng ứng dụng có thể thiết lập, duy trì và giải phóng các cuộc gọi hoặc các phiên truyền thông. Các phiên truyền thông có thể là điện thoại hội nghị, học từ xa, điện thoại Internet và các ứng dụng tương tự khác. SIP có thể phục vụ cho các thành viên cả các phiên đơn hướng và đa hướng (Unicast và Multicast). Người bắt đầu không nhất thiết phải là một thành viên của phiên truyền thông. Phương tiện và các thành viên có thể được bổ sung vào phiên hiện tại. SIP cũng có thể được dùng để bắt đầu các phiên cũng như dùng để mời các thành viên tới phiên hội thoại mà được thông báo và được thiết lập bởi các phương tiện khác. SIP hỗ trợ các dịch vụ ánh xạ tên và các dịch vụ gián tiếp một cách trong suốt. Vì thế nó cho phép thi hành một cách đầy đủ các dịch vụ trên ISDN, mạng thoại thông minh và hỗ trợ các cuộc gọi di động của người dùng có địa chỉ không cố định.

SIP hỗ trợ 5 dịch vụ trong việc thiết lập và kết thúc các phiên truyền thông:

- Định vị User: Xác định vị trí của User tiến hành hội thoại.
- Năng lực User: Xác định phương tiện và tham số tương ứng trong hội thoại.
- Xác định những User sẵn sàng tham gia hội thoại.
- Thiết lập các tham số cần thiết cho cuộc gọi.
- Điều khiển cuộc gọi: bao gồm cả quá trình truyền và kết thúc cuộc gọi.

SIP có thể cũng tạo ra cuộc gọi hội nghị (cuộc gọi gồm nhiều người tham gia) bằng cách sử dụng một khối điều khiển đa điểm (MCU).

SIP là một phần trong bộ giao thức chuẩn cho truyền dòng tin đa phương thức do IETF khuyến nghị như RSVP (Resource reSerVation Protocol - Giao thức đặt trước tài nguyên), RTP (Realtime Transport Protocol - Giao thức truyền tải thời gian thực), RTSP (RealTime Streaming Protocol - Giao thức phân phối dòng tin thời gian thực), SDP (Session Description Protocol - Giao thức mô tả phiên). Tuy nhiên, SIP hoạt động độc lập với các giao thức trên.

SIP cũng có thể sử dụng kết hợp với các giao thức báo hiệu và thiết lập cuộc gọi khác. Theo cách đó, một hệ thống đầu cuối dùng SIP để xác định địa chỉ hợp lệ của một hệ thống và khi đó giao thức từ một địa chỉ gửi đến là giao thức độc lập. Ví dụ: SIP có thể dùng để chỉ ra rằng người tham gia có thể thông qua H.323, cổng H.245, địa chỉ người dùng rồi dùng H.225 để thiết lập cuộc gọi.

5.2.2 Các tính năng của SIP

Giao thức SIP được thiết kế với những tiêu chí hỗ trợ tối đa cho các giao thức khác đã ra đời trước đó. Giao thức SIP nó được tích hợp với các giao thức đã có của tổ chức IETF, nó có khả năng mở rộng, hỗ trợ đầu cuối và với SIP thì việc cung cấp dịch vụ mới trở nên dễ dàng và nhanh chóng khi triển khai. Cụ thể ta xét SIP với 4 tính năng sau:

- Tích hợp với các giao thức đã có của IETF
- Đơn giản và có khả năng mở rộng
- Hỗ trợ tối đa sự di động của đầu cuối
- Dễ dàng tạo tính năng mới cho dịch vụ và dịch vụ mới

Tích hợp với các giao thức đã có của IETF

Các giao thức khác của IETF có thể sử dụng để xây dựng những ứng dụng SIP. SIP có thể hoạt động cùng với nhiều giao thức như:

- RSVP (Resource reSerVation Protocol): giao thức đặt trước tài nguyên.
- RTP (Realtime Transport Protocol): giao thức truyền tải thời gian thực.
- RTSP (RealTime Streaming Protocol): giao thức phân phối dòng tin thời gian thực.
- SAP (Session Advertisement Protocol): giao thức quảng cáo trong phiên kết nối.
- SDP (Session Description Protocol): giao thức mô tả các phiên kết nối đa phương tiện.
- MIME (Multipurpose Internet Mail Extension): giao thức thư điện tử nhiều mục đích.
- HTTP (HyperText Transfer Protocol): giao thức để truy nhập trang Web.
- COPS (Common Open Policy Service): dịch vụ giải quyết mở chung.
- OSP (Open Settlement Protocol): giao thức thỏa thuận mở.

Đơn giản và có khả năng mở rộng

SIP có rất ít bản tin, không có các chức năng thừa nhưng SIP có thể sử dụng để thiết lập những phiên kết nối phức tạp như hội nghị... Đơn giản, gọn nhẹ, dựa trên khuôn dạng văn bản (text), SIP là giao thức ra đời sau và đã khắc phục được điểm yếu của nhiều giao thức trước đây.

Các phần mềm Proxy Server, Registrar Server, Redirect Server, Location Server... có thể chạy trên các máy chủ khác nhau và việc cài đặt thêm máy chủ hoàn toàn không

ảnh hưởng đến các máy chủ đã có. Chính vì thế hệ thống chuyển mạch SIP có thể dễ dàng nâng cấp.

Hỗ trợ tối đa sự di động của đầu cuối

Do có Proxy Server, Registrar Server và Redirect Server hệ thống luôn nắm được địa điểm chính xác của thuê bao. Thí dụ thuê bao với địa chỉ SIP: hvcnbcvt@vnpt.vn có thể nhận được cuộc gọi thoại hay thông điệp ở bất cứ địa điểm nào qua bất cứ đầu cuối nào như máy tính để bàn, máy xách tay, điện thoại SIP... Với SIP rất nhiều dịch vụ di động mới được hỗ trợ như Presence (biết trạng thái hiện thời của đầu cuối) và Call forking (phân nhánh cuộc gọi).

Dễ dàng tạo tính năng mới cho dịch vụ và dịch vụ mới

Là giao thức khởi tạo phiên trong mạng chuyển mạch gói, SIP cho phép tạo ra những tính năng mới hay dịch vụ mới một cách nhanh chóng. Ngôn ngữ xử lý cuộc gọi CPL (Call Processing Language) và giao diện cổng chung CGI (Common Gateway Interface) là một số công cụ để thực hiện điều này. SIP hỗ trợ các dịch vụ thoại như: chờ cuộc gọi (Call Waiting), chuyển tiếp cuộc gọi (Call Forwarding), chặn cuộc gọi (Call Blocking), hỗ trợ thông điệp thống nhất...

5.3. HOẠT ĐỘNG CỦA SIP

Trong hội thoại SIP, mỗi bên tham gia (bên chủ gọi và bên bị gọi) được gắn một địa chỉ SIP hay còn gọi là SIP URL (SIP Uniform ResourceLocator). Người sử dụng phải đăng ký vị trí của họ với SIP Server. Để tạo một cuộc gọi SIP, phía chủ gọi định vị tới máy chủ thích ứng và sau đó gửi một yêu cầu SIP. Hoạt động SIP thường xuyên nhất là lời mời các thành viên tham gia hội thoại. Thành phần đăng ký đóng vai trò tiếp nhận các yêu cầu đăng ký từ UA (User Agent) và lưu trữ các thông tin này tại một dịch vụ phi SIP (Non-SIP).

5.3.1 Địa chỉ SIP (SIP Address)

Các thành viên tham gia hội thoại được định danh bởi một địa chỉ SIP gọi là SIP URL. SIP URL được dùng trong các bản tin SIP để thông báo về nơi gửi (From), đích hiện thời (Request URI) và nơi nhận cuối cùng (To) của một yêu cầu SIP và chỉ rõ địa chỉ gián tiếp. Một SIP URL có thể gắn vào một trang Web hoặc những siêu liên kết (Hyperlink) khác để thông báo rằng người dùng hoặc dịch vụ có thể gọi thông qua SIP.

Cú pháp của SIP URL được viết bằng ngôn ngữ BNF (Backus Naur Form). SIP URL có dạng:

```
SIP-URL = "sip:" [userinfo"@" ]hostport
                    url-parameters[ headers ]
    userinfor =user[ ":" password ]
          user = *(unreserved | escaped | "&" | "=" | "+" | "$" | ",")
          password = *(unreserved | escaped | "&" | "=" | "+" | "$" | ",")
    hostport = host[ ":" port ]
          host = hostname | IPv4address
               hostname = *( domainlabel ".") toaplabel ["."]
                    domainlabel = alphanum | alphanum * (alphanum | "-") alphanum
                    toaplabel = alpha | alpha *(alphanum | "-") alphanum
                   IPv4address = 1*digit "."1*digit "."1*digit "."1*digit
          port = *digit
    url-parameters = *(";"url-parameter )
          url-parameter = transport-param | user-param | method-param | ttl-param |
                                maddr-param | other-param
                transport-param = "transport=" ("udp" | "tcp" )
                user-param = "user = "( "phone " | "ip" )
                method-param = "method=" Method
                ttl-param = "ttl=" ttl
                      ttl = 1*3digit; 0 to 255
                maddr-param = "maddr=" host
                other-param = ( token |(token "=" (token | quoted-string )))
    headers = "? "header * ( "&" header )
          header = hname"=" hvalue
                hname = 1*uric
                hvalue = *uric
                      uric = reserved | unreserved | escaped
                        reserved= ";"| "/"| "?"| ":"| "@"|"&"|"="|"+"|"$"|","
    digits =1*DIGIT
```

Trong SIP URL không phân biệt chữ hoa hay chữ thường. SIP URL cho thuê bao thoại có dạng:

telephone-subscriber = global-phone-number | local-phone-number

global-phone-number = "+"1*phonedigit[isdn-subaddress][post-dial]

local-phone-number = 1*(phonedigit | dtmf-digit | pause-character)
[isdn-subaddress][post-dial]

isdn-subaddress = ";isub"1*phonedigit

post-dial = ";postd="1*(phonedigit | dtmf | pause-character)

phonedigit =DIGIT | visual-separator

visual-separator = "-" |"."

pause-character = one-second-pause | wait-for-dial-tone

one-second-pause ="p"

wait-for-dial-tone = "w"

dtmf-digit = "*"| "#"| "A" | "B" | "C" | "D"

Bảng 5.2 Thống kê các vị trí các thành phần SIP URL được sử dụng.

Bảng 5.2: Vị trí các thành phần SIP URL được sử dụng

	Default	Req-URI	To	From	Contact	External
User	- -	x	x	x	x	x
Password	- -	x	x	x	x	x
Host	Madatory	x	x	x	x	x
Port	5060	x	x	x	x	x
User-param	IP	x	x	x	x	x
Method	INVITE				x	x
Maddr-param	- -				x	x
Ttl-param	1				x	x
Transp-param	- -				x	x
Hearders	- -				x	x

User: Nếu máy chủ là một cổng thoại Internet, trường User có thể được dùng để mã hóa số điện thoại, chú thích về thuê bao thoại. Số điện thoại là trường hợp đặc biệt của tên User và không thể phân biệt bởi BNF. Như vậy một thông số URL khác User, được thêm vào để phân biệt số điện thoại từ tên User.

- *Password:* Cấu trúc SIP có thể dùng dạng "user password" trong trường thông tin người dùng.

- *Host:* Địa chỉ Email URL và RFC 822 yêu cầu địa chỉ số học của Host phải được đặt trong dấu ngoặc vuông trong khi số Host không có ngoặc được dùng cho tất cả các URL khác.

- *Port:* Số cổng để gửi một yêu cầu đi.

- *Thông số URL*: SIP URL có thể định nghĩa những thông số đặc trưng của yêu cầu. Thông số URL được thêm vào sau thành phần Host và được ngăn cách bởi một dấu chấm phẩy ";". Thông số truyền chỉ ra giao thức truyền tin là UDP hay TCP. UDP được sử dụng khi tính đến các thông số truyền không rõ ràng. Thông số "maddr" cung cấp địa chỉ của Server sẽ liên lạc với User (quan trọng hơn là cung cấp địa chỉ cho trường Host). Địa chỉ này là một địa chỉ đa hướng điển hình nhưng cũng có thể là một địa chỉ của Server dự phòng (Backup - Server). Thông số "ttl" (time to live) xác định thời gian sống của gói tin quảng bá UDP và chỉ được sử dụng khi "maddr" là một địa chỉ quảng bá và giao thức truyền là UDP. Ví dụ: để diễn tả một cuộc gọi j.doe@big.com sử dụng multicast 239.255.255.1 với ttl là 15, SIP URL sẽ là SIP: j.doe@big.com; maddr = 239.255.255.1; ttl = 15.

Các thông số "transport", "maddr", "ttl" không được dùng trong trường tiêu đề "To" và "From", "Request-URI", chúng sẽ bị loại bỏ nếu như xuất hiện.

Header: tiêu đề của yêu cầu SIP có thể được xác định bằng dấu "?" trong SIP URL. Hname đòi hỏi hvalue là một thân bản tin của yêu cầu INVITE. Header không được dùng trong trường tiêu đề "To", "From" và "Request-URI". "Hname" và "hvalue" là mã tương ứng với tên tiêu đề và giá trị tiêu đề của SIP. Tất cả những ký tự riêng trong tên tiêu đề và value (giá trị) đều phải loại bỏ.

- *Method:* chỉ thị của yêu cầu SIP được đặc trưng bởi thông số "method". Thông số này cũng không được dùng trong trường tiêu đề "From" và "To", nếu có thì chúng sẽ bị loại bỏ.

5.3.2 Giao dịch SIP (SIP Transaction)

Khi có địa chỉ IP của SIP Server thì yêu cầu sẽ được gửi đi theo tầng vận chuyển giao thức TDP hay UDP. Client gửi một hay nhiều yêu cầu SIP tới SIP Server và nhận các đáp ứng từ Server. Một yêu cầu cùng với những đáp ứng cho những yêu cầu đó tạo nên một giao dịch SIP. Tất cả các đáp ứng cho một yêu cầu phải chứa cùng các giá trị

trong trường Call-ID, CSeq, To và From. Điều đó làm cho các đáp ứng phù hợp với các yêu cầu. Mỗi cuộc gọi trong SIP được định danh bởi một trường định danh cuộc gọi (Call-ID).

Một yêu cầu phải cần có thông tin gửi đi từ đâu (From) và tới đâu (To). Trường From và To đều có cấu trúc theo khuôn dạng SIP-URL. Trường CSeq lưu trữ thông tin về phương thức sử dụng trong phiên, trường CSeq có dạng:

CSeq = "CSeq":"DIGIT Method"

Trong đó DIGIT là số nguyên không dấu 32 bit.

Nếu giao thức TCP được sử dụng, các yêu cầu và đáp ứng trong một giao dịch SIP đơn giản đều được truyền qua cùng một kết nối TCP. Một số yêu cầu SIP từ cùng một Client đến cùng một Server có thể dùng cùng một kiểu nối TCP hoặc có thể dùng một kiểu kết nối mới cho mỗi yêu cầu. Nếu Client gửi yêu cầu thông qua giao thức UDP đơn hướng, các đáp ứng sẽ được gửi đến các địa chỉ nằm trong trường tiêu đề "Via" của đáp ứng. Nếu yêu cầu được gửi qua giao thức UDP đa hướng thì các đáp ứng sẽ được đưa đến cùng một địa chỉ đa hướng và cổng đích. Khuôn dạng và hoạt động của bản tin SIP không phụ thuộc vào giao thức truyền.

5.3.3 Lời mời SIP

Một lời mời SIP thành công gồm hai yêu cầu INVITE và ACK. Yêu cầu INVITE thực hiện lời mời một thành viên tham gia hội thoại. Khi phía bị gọi đồng ý tham gia, phía chủ gọi xác nhận đã nhận một bản tin đáp ứng bằng cách gửi đi một yêu cầu ACK. Nếu phía chủ gọi không muốn mời thành viên tham gia cuộc gọi nữa nó sẽ gửi yêu cầu BYE thay cho ACK.

Thông điệp INVITE chứa thành phần mô tả phiên (SDP) và phương thức tiến hành trao đổi ứng với phiên đó. Với các phiên đa hướng, phần mô tả phiên liệt kê kiểu và khuôn dạng của các phương tiện (Media) để phân phối cho phiên hội thoại. Với một phiên đơn hướng, phần mô tả phiên liệt kê kiểu và khuôn dạng của các phương tiện mà phía chủ gọi muốn sử dụng và nơi những dữ liệu muốn gửi đi.

Trường hợp máy chủ ủy quyền (Proxy Server):

- Proxy Server tiếp nhận lời mời INVITE
- Proxy Server tra cứu thông tin ở dịch vụ định vị ngoài SIP
- Proxy Server nhận thông tin để tạo ra địa chỉ chính xác

- Proxy Server tạo lại INVITE trong trường Request URI và chuyển tiếp

- UAS thông báo bị gọi

- Proxy Server nhận đáp ứng chấp nhận 200 OK từ UAS

- Proxy Server trả về kết quả thành công cho chủ gọi

- Chủ gọi gửi thông báo xác nhận ACK

- Yêu cầu xác nhận được chuyển tiếp qua Proxy Server

Chú ý: Một ACK có thể được gửi trực tiếp đến User được gọi qua Proxy. Tất cả các yêu cầu và đáp ứng phải có cùng Call-ID (trường danh định cuộc gọi).

Trường hợp máy chủ chuyển đổi địa chỉ (Redirect Server):

- Proxy Server tiếp nhận lời mời INVITE

- Liên lạc với dịch vụ định vị

- Trả lời địa chỉ chủ gọi

- Chủ gọi gửi thông báo xác nhận ACK đến Proxy Server

- Chủ gọi tạo một yêu cầu mới cùng một Call-ID nhưng có CSeq cao hơn tới địa chỉ trả lời bởi máy chủ đầu tiên

- Bị gọi gửi đáp ứng chấp nhận 200 OK

- Chủ gọi gửi thông báo xác nhận ACK

5.3.4 Quá trình định vị tới máy chủ SIP

Khi một Client muốn gửi đi một yêu cầu, Client sẽ gửi bản tin yêu cầu đó tới máy chủ ủy quyền SIP (SIP Proxy Server) (như trong HTTP), hoặc tới địa chỉ IP và cổng tương ứng trong địa chỉ của yêu cầu SIP (Request-URI). Trường hợp đầu, yêu cầu được gửi tới SIP Proxy Server không phụ thuộc vào địa chỉ của yêu cầu đó là như thế nào. Với trường hợp sau, Client phải xác định giao thức, cổng và địa chỉ IP của Server mà yêu cầu được gửi đến. Một Client thực hiện các bước tiếp theo để có được những thông tin này. Tại mỗi bước, trừ các trạng thái khác, Client cố gắng liên lạc với Server theo số cổng được chỉ ra trong địa chỉ yêu cầu SIP (Request-URI). Nếu không có số cổng nào chỉ ra trong Request-URI, Client sẽ sử dụng địa chỉ cổng là 5060. Nếu Request-URI chỉ rõ là sử dụng giao thức TCP hay UDP, Client sẽ làm việc với Server theo giao thức đó. Nếu không có giao thức nào được chỉ ra thì Client cố gắng dùng giao thức UDP (nếu

không hỗ trợ TCP). Nếu mọi nỗ lực không thành công hay Client không được hỗ trợ UDP mà chỉ được hỗ trợ TCP thì nó sẽ cố gắng sử dụng giao thức TCP cho hoạt động của mình.

Client cố gắng tìm một hay nhiều địa chỉ cho SIP Server nhờ thực hiện DNS (Domain Name System - Hệ thống tên miền). Các thủ tục như sau:

1. Nếu địa chỉ Host trong địa chỉ Request-URI (Uniform Resource Identifier) là một địa chỉ IP thì Client làm việc với Server bằng địa chỉ được đưa ra. Nếu đó không phải là một địa chỉ IP, Client thực hiện bước tiếp theo.

2. Client đưa ra câu hỏi tới máy chủ DNS về bản ghi địa chỉ cho địa chỉ Host trong địa chỉ Request-URI. Nếu máy chủ DNS không đưa ra bản ghi địa chỉ, Client sẽ kết thúc hoạt động, có nghĩa nó không thực hiện được việc định vị máy chủ. Nhờ bản ghi địa chỉ, sự lựa chọn tiếp theo cho giao thức mạng của Client có nhiều khả năng thành công hơn.

Không có một quy định bắt buộc nào về cách lựa chọn tên Host cho máy chủ SIP. Người dùng được khuyến khích dùng tên của SIP Server là tên miền SIP (SIP.domainname). Theo truyền thống, như chỉ ra trong RFC 2219, người dùng có thể chỉ biết một địa chỉ email thay vì biết tất cả địa chỉ đầy đủ của SIP URL cho bị gọi. Tuy nhiên, trong nhiều trường hợp, việc thực hiện có thể được thực hiện một cách chắc chắn hơn (khi làm việc với SIP Server) đối với miền đó nhờ cấu trúc của SIP URL được lấy từđịa chỉ email đó với phần tiền tố tên Host là "sip". Một quá trình thực hiện thành công là quá trình có một bản ghi được chứa trong phần trả lời và Server làm việc ở một trong những địa chỉ chứa trong trả lời đó.

5.3.5 Định vị người dùng

User được gọi có thể di chuyển giữa các hệ thống đầu cuối khác nhau tại các thời điểm khác nhau. Những vị trí đó được đăng ký với SIP Server. Một Location Server (máy định vị) có thể sử dụng một hay nhiều giao thức để xác định đầu cuối mà User có thể làm việc tới. Một Location Server có thể trả lại vài vị trí mà User đã đăng ký đồng thời tại nhiều Host hoặc bởi Location Server có lỗi. SIP Server sẽ tổng kết các kết quả để đưa ra danh sách các vị trí. Đối với các loại SIP Server thì hoạt động sau khi nhận được các vị trí khác nhau là khác nhau. Một SIP Redirect Server sẽ trả lại danh sách địa chỉ cho Client với tiêu đề Contact. Một SIP Proxy Server có thể liên tục thử tuần tự các địa chỉ cho đến khi cuộc gọi thành công hay User được gọi từ chối cuộc gọi. Nếu Proxy Server gửi một yêu cầu SIP, nó phải bổ sung thêm địa chỉ của chính nó vào phần đầu danh sách của hướng đi trong tiêu đề "Via". Dấu hiệu "Via"đảm bảo rằng bản tin trả lời

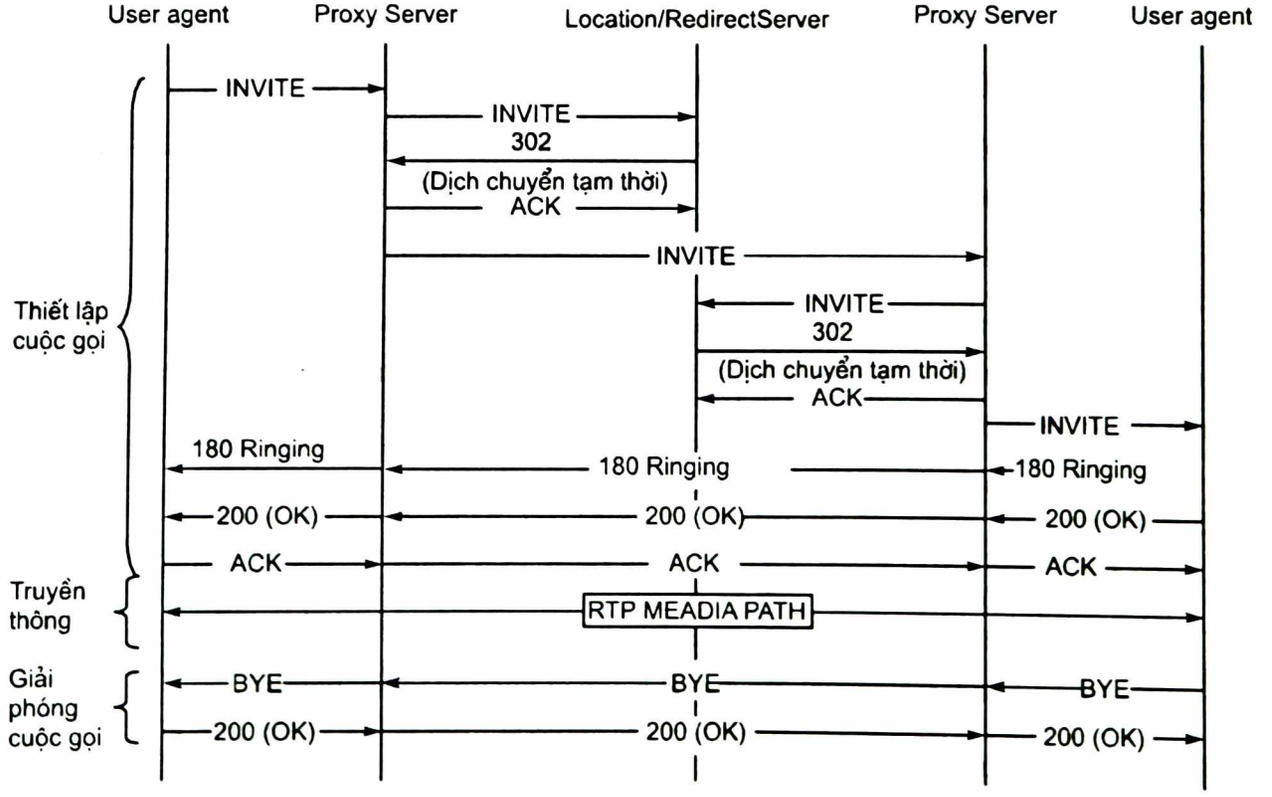
có thể lấy ra từ cùng một hướng. Ở hướng trả lời, mỗi Host phải gỡ bỏ tiêu đề "Via" của chính nó, vì thế thông tin được truyền nội bộ sẽ "ẩn" (hide) đối với phía bị gọi và mạng bên ngoài. Proxy Server phải kiểm tra xem nó có phát yêu cầu tới danh sách Host (Host list) chứa trong các tham số Via rent-by, Via Received hay Via maddr.

5.3.6 Thay đổi một phiên hiện tại

Trong một vài trường hợp, cần phải thay đổi các thông số của phiên hội thoại hiện tại. Việc đó được thực hiện bởi việc phát lại các yêu cầu INVITE. Các yêu cầu INVITE đó có cùng trường Call-ID nhưng có trường tiêu đề và trường bản tin khác với yêu cầu ban đầu để mang thông tin mới. Các bản tin INVITE đó phải có chỉ số CSeq cao hơn các yêu cầu trước. Ví dụ: có hai thành viên đang hội thoại và muốn có thêm một người thứ ba tham gia. Một trong hai thành viên sẽ mời thành viên thứ ba tham gia với một địa chỉ đa hướng (Multicast) mới và đồng thời gửi một bản tin INVITE đến thành viên thứ hai với trường miêu tả phiên đa hướng nhưng có trường Call-ID cũ.

5.3.7 Thiết lập và giải phóng cuộc gọi SIP

Quá trình thiết lập và giải phóng cuộc gọi SIP được mô tả trong hình 5.2:



Hình 5.2: Quá trình thiết lập và giải phóng một cuộc gọi SIP

Trong mạng SIP quá trình thiết lập và giải phóng một phiên kết nối thường gồm 6 bước sau:

1. Đăng ký, khởi tạo và định vị dịch vụ đầu cuối.

2. Xác định phương tiện của cuộc gọi, tức đưa ra một miêu tả phiên mà đầu cuối được mời tham dự.

3. Xác định mong muốn của đầu cuối bị gọi, trả lời hay không. Phía bị gọi phải gửi bản tin xác nhận chấp nhận cuộc gọi hay từ chối.

4. Thiết lập cuộc gọi.

5. Thay đổi hay điều khiển cuộc gọi (thí dụ như chuyển cuộc gọi).

6. Giải phóng cuộc gọi.

5.4 BẢN TIN SIP

SIP là giao thức dạng văn bản sử dụng bộ ký tự ISO 10646 trong mã hóa UTF-8 (RFC 2279). Điều này tạo cho SIP tính linh hoạt, mở rộng và dễ thực thi các ngôn ngữ lập trình cấp cao như Java,... Cú pháp của SIP gần giống với giao thức HTTP, nó cho phép dùng lại mã và đơn giản hóa sự liên kết của các máy chủ SIP với các máy chủ Web. Tuy nhiên, lưu ý rằng SIP không phải là một dạng mở rộng của HTTP. Khác với HTTP, SIP có thể sử dụng giao thức UDP.

Bản tin SIP có thể là một yêu cầu từ Client tới một Server hay đó là một đáp ứng từ Server gửi lại Client.

SIP-message = Request/ Response

Cả hai bản tin yêu cầu và đáp ứng đều có một khuôn dạng chung. Cả hai loại đều gồm một dòng khởi đầu (start-line), một hay nhiều tiêu đề, một chỉ thị kết cuối tiêu đề và một tùy chọn thân bản tin:

generic-message = start-line * message-header

CRLF

[message-body]

+ start-line = Request-line/ Status-line (general-header)

+ message-header=Request-header/ Response-header/ Entity-header

5.4.1 Bản tin Request

Bản tin Request (yêu cầu) có khuôn dạng như sau:

Request = Request-line *(General-header/ Request-header/ Entity-header)

CLRF (ký tự xuống dòng)

[message-body]

Ta thấy bản tin Request gồm hai phần cơ bản: Request-line (dòng yêu cầu) và header (phần tiêu đề):

Request-line

Thành phần Request-line bắt đầu bởi một cờ chỉ thị, tiếp theo là Request-URI và phiên bản sử dụng, kết cuối bởi CRLF. Các thành phần được phân cách bởi ký tự **SP**.

Request-line = Method **SP** Request-URI **SP** SIP-Version CRLF

1) Method (Chỉ thị SIP)

Các chỉ thị bắt đầu không được hỗ trợ bởi Proxy Server hay Redirect Server. Với SIP người ta định nghĩa 6 chỉ thị sau:

Method = INVITE/ACK/OPTION/ BYE/CANCEL/REGISTER

a) INVITE: Chỉ thị INVITE thông báo rằng User hoặc dịch vụ được mời tham gia vào một phiên hội thoại. Một Server sẽ tự động trả lời một lời mời tham gia hội thoại nếu User đã sẵn sàng tham gia bằng đáp ứng 200 OK.

Nếu như một UA (User Agent) nhận được một yêu cầu INVITE cho "Call-leg" với số CSeq cao hơn các INVITE trước có cùng Call-ID, nó sẽ kiểm tra các từ định danh phiên bản. Nội dung của phần miêu tả phiên sẽ được xem xét nếu nó muốn thay đổi, UA phải xem xét các trường tiêu đề cho việc thay đổi, UA phải cập nhật những trạng thái nội bộ hoặc những thông tin phát ra như kết quả tiêu đề đó. Nếu miêu tả phiên thay đổi, UAS phải điều chỉnh lại các thông số của phiên hội thoại cho phù hợp sau khi yêu cầu User xác nhận. Chỉ thị này được đưa ra bởi SIP Proxy, Redirect Server và UAS cũng như Client.

b) ACK: Yêu cầu ACK xác nhận Client đã nhận được đáp ứng cuối cùng cho yêu cầu INVITE (ACK chỉ sử dụng cho yêu cầu INVITE). Khi UAC chấp nhận đáp ứng 2xx, tất cả các đáp ứng cuối cùng khác của Proxy đầu tiên hay của UAC đều nhận được trả lời. Trường "Via" được đưa vào Host để phát ra yêu cầu ACK sau khi UAC nhận được một đáp ứng 2xx hoặc Proxy đầu tiên nhận được một đáp ứng non-2xx.

Yêu cầu ACK có thể chứa một thân bản tin (message body) với phần miêu tả phiên cuối cùng cho User được gọi. Nếu như thân bản tin ACK rỗng, User được gọi sẽ sử dụng phần miêu tả phiên trong yêu cầu INVITE.

Một Proxy Server nhận một yêu cầu ACK sau khi gửi đi các đáp ứng 3xx, 4xx, 5xx hay 6xx phải quyết định ACK là của nó hay là của các UA hoặc Proxy Server phía bên kia. Quyết định đó dựa trên việc xem xét thẻ địa chỉ trong trường "To". Nếu thẻ của trường "To" trong ACK phù hợp với thẻ địa chỉ trong trường tiêu đề "To" của đáp ứng và các trường tiêu đề "From", "Call-ID", "CSeq" của đáp ứng phù hợp với các trường của ACK thì chứng tỏ rằng ACK là dành cho Proxy Server.

Chỉ thị này phải được đưa ra bởi SIP Proxy, Redirect Server, UA cũng như Client.

c) OPTION: Chỉ thị OPTION dùng để hỏi về khả năng của SIP Server. Nếu một Server có khả năng liên lạc với User thì có thể đáp ứng lại yêu cầu OPTION bằng một tập hợp các khả năng của nó. Chỉ thị này được đưa ra bởi SIP Proxy, Redirect Server, UA, Register, Client.

d) BYE: UAC sử dụng chỉ thị BYE để thông báo cho Server rằng nó muốn giải phóng cuộc gọi. Nếu yêu cầu INVITE chứa một tiêu đề "Contact", User được gọi phải gửi yêu cầu BYE tới địa chỉ trong trường "Contact". Chỉ thị này phải được đưa ra bởi một Proxy Server và không được phát ra bởi Redirect Server và UAS.

e) CANCEL: Yêu cầu CANCEL được dùng để hủy bỏ một yêu cầu sắp thực hiện với cùng giá trị trong các trường Call-ID, From, To và CSeq của yêu cầu đó.

UAC hay Proxy Client có thể phát ra một yêu cầu CANCEL tại mọi lúc. Proxy trong trường hợp đặc biệt có thể gửi một yêu cầu CANCEL tới đích mà không trả lại một đáp ứng cuối cùng sau khi nó đã nhận được các đáp ứng 2xx hay 6xx cho một hay nhiều yêu cầu tìm kiếm song song (Parallel Search).

Các trường Call-ID, CSeq, To, From trong CANCEL đều giống như trong yêu cầu ban đầu (yêu cầu mà nó muốn hủy bỏ). Tuy nhiên, để Client phân biệt được các đáp ứng cho yêu cầu CANCEL với các đáp ứng cho yêu cầu ban đầu thì thành phần CSeq Method sẽ được thiết lập trong CANCEL.

Khi một UAS nhận được yêu cầu CANCEL, nó không phải phát ra một đáp ứng 2xx cho yêu cầu hủy bỏ ban đầu.

Redirect Server và UAS khi nhận được một yêu cầu CANCEL sẽ đáp ứng bằng một trạng thái 200 nếu như tồn tại giao dịch SIP hoặc trạng thái 481 nếu không tồn tại giao dịch.

Chỉ thị này được phát ra bởi tất cả các loại SIP Proxy Server.

f) REGISTER: Chỉ thị REGISTER được dùng để đăng ký danh sách địa chỉ của User trong trường tiêu đề To với SIP Server. UA có thể đăng ký với một Local Server

lúc khởi động bằng cách gửi đi một yêu cầu REGISTER tới tất cả các SIP Server có địa chỉ đa hướng là "Sip. Mcasr. net".

Client tránh việc gửi đi một đăng ký mới cho tới khi nó nhận được đáp ứng trả lại từ Server cho đăng ký trước đó. Client có thể đăng ký với nhiều dịch vụ khác nhau và nó nhất thiết phải sử dụng các giá trị Call-ID khác nhau.

Ý nghĩa của trường tiêu đề yêu cầu REGISTER được định nghĩa dưới đây. Ta có thể định nghĩa địa chỉ bản ghi (Address of record) như địa chỉ SIP.

To: Trường tiêu đề To chứa địa chỉ bản ghi trong suốt quá trình bản đăng ký được tạo ra và cập nhật.

From: Trường tiêu đề From chứa địa chỉ bản ghi của người muốn đăng ký.

Request-URI: Trường Request-URI chỉ ra đích của yêu cầu đăng ký. Nói chung các trường trong miền Request-URIvà To đều có giá trị giống nhau.

Call-ID: Tất cả các đăng ký từ cùng một User đều sử dụng một giá trị tiêu đề Call-ID.

Cseq: Các đăng ký với cùng một giá trị Call-ID phải được tăng chỉ số CSeq lên một giá trị.

Contact: Yêu cầu đăng ký có thể chứa một trường tiêu đề Contact. Nếu như yêu cầu không chứa tiêu đề Contact, bản đăng ký sẽ giữ nguyên không đổi.

Server có thể trả lại danh sách đăng ký hiện tại bằng đáp ứng 200 OK.

2) Request-URI

Trường Request-URI có khuôn dạng theo SIP URL. Nó thông báo cho User hoặc dịch vụ về địa chỉ hiện tại. Khác với trường "To", Request-URI có thể được ghi lại bởi Proxy. Khi sử dụng như một Request-URI, SIP URL phải chứa các tham số transport-param, maddr-param, ttl-param và các thành phần tiêu đề. Một Server khi nhận được một SIP URL (địa chỉ SIP) với những thành phần đó sẽ hủy bỏ chúng trước khi tiến hành xử lý.

Điển hình là, UAS thiết lập trường Request-URI và trường "To" trong cùng một SIP URL coi như không thay đổi trong một khoảng thời gian dài. Tuy nhiên, nếu như UAC dành cho phía bị gọi nhiều hướng trực tiếp, từ trường tiêu đề Contact của một đáp ứng cho yêu cầu trước đó, trường "To" sẽ vẫn chứa thông số "long-term", địa chỉ công cộng trong khi Request-URI có thể thiết lập các địa chỉ dự phòng. Proxy Server và

Redirect Server có thể dùng những thông tin chứa trong trường Request-URI và trường tiêu đề của yêu cầu để quản lý các yêu cầu và ghi lại trường Request-URI.

Host-post của Request-URI điển hình phù hợp với một trong các "Host name" của Server nhận. Nếu không, Server phải ủy quyền cho yêu cầu tới địa chỉ "Indicate" hoặc trả lại đáp ứng 404 (Not found: không tìm thấy) nếu nó không muốn hay không thể thực hiện được. Ví dụ, Request-URI và "Server host name" có thể không phù hợp trong trường hợp tường lửa xử lý cuộc gọi đi. Cách hoạt động đó giống như của HTTP Proxy.

Nếu SIP Server nhận được một yêu cầu với một URI đưa ra một cấu trúc khác hơn SIP mà Server đó không hiểu được, Server phải gửi lại một đáp ứng 400 (Bad Request). Nó phải thực hiện việc đó ngay cả khi trường tiêu đề "To" chứa một cấu trúc mà nó không hiểu.

3) SIP Version (phiên bản SIP)

Phiên bản SIP (SIP version) là các bản SIP được đưa ra các lần khác nhau. Cả hai bản tin Request và Response đều chứa phiên bản của SIP được sử dụng. Hiện tại phiên bản SIP là 2.0.

4) Option tag

Thẻ tùy chọn (Option tag) được dùng để chỉ định các lựa chọn mới trong SIP. Các thẻ tùy chọn được sử dụng trong trường Unsupport và Require.

Cú pháp: Option-tag = token

Các tùy chọn mới (new options) được đăng ký bởi tổ chức IANA (Internet Assigned Number Authority).

- Khi đăng ký thì cần phải có các thông tin sau:
- Tên và mô tả của các Option. Tên có độ dài không quá 20 ký tự.
- Thông báo về tổ chức có quyền thay đổi Option.
- Thông tin liên lạc (hòm thư hay địa chỉ Email).
- Đăng ký phải được gửi tới địa chỉ: iana@Lana.crg.

Tiêu đề của bản tin Request

Trong khuôn dạng bản tin Request ta thấy có 3 dạng trường tiêu đề của bản tin Request: General-header, Request-header và Entity-header

General-header=Acept

| Acept-Encoding

| Acept-Language

| Call-ID

| Contact

| CSeq

| Date

| Encryption

| Expires

| From

| Record-route

| Timestamp

| To

| Via

Request-header= Authorization

| Contact

| Hide

| Max-Forwards

| Organization

| Priority

| Proxy-Authorization

| Proxy- Require

| Route

| Require

| Respone-Key

| Subject

| User-Agent

Entity-header= Content-Encoding

| Content-Length

| Content-Type

5.4.2 Bản tin đáp ứng (Response)

Sau khi nhận và thông dịch một bản tin yêu cầu, phía nhận thực hiện trả lời bằng một bản tin đáp ứng. Khuôn dạng bản tin Response như sau:

Response = Status-line *(General-header/ Response-header/ Entity-header)

CLRF

[message-body]

Sau đây ta đi phân tích Status-line và tiêu đề của bản tin Response:

Status-line

Thành phần đầu tiên của bản tin đáp ứng là status-line, nó bao gồm một phiên bản giao thức sử dụng, tiếp theo là con số mã trạng thái (status-code) kết hợp với reason-phrase, kết cuối bởi CRLF. Giữa các thành phần phân chia bởi ký tự **SP**. Cấu trúc của Status-line có dạng như sau:

Status-line = SIP-Version**SP** status-code **SP** Reason-phrase CRLF

1) SIP Version

Phiên bản SIP (SIP version) là các bản SIP được đưa ra các lần khác nhau. Cả hai bản tin Request và Response đều chứa phiên bản của SIP được sử dụng SIP Version. Hiện tại phiên bản SIP là 2.0.

2) Status-code và Reason-phrase

Status-code là một mã kết quả nguyên gồm 3 chữ số, chỉ ra kết quả của việc cố gắng thực hiện và mức độ thỏa mãn yêu cầu. Status-Phrase thì dùng để đưa ra một lờigiải thích ngắn cho Status-code. Status-code mục đích là để sử dụng cho Server còn Reason-phrase là dùng cho User. Client không thể yêu cầu hiển thị hay kiểm tra Reason-phrase.

Reason-Phrase =	Informational	Tương ứng	Status-code =	1xx
	/Success	....	.....	/2xx
	/Redirection	....	.....	/3xx
	/Client-Error	....	.....	/4xx
	/Server-Error	....	.....	/5xx
	/Global Failure	....	.....	/6xx
	/Extension-Code			

Trong đó: Extension-code = 3 chữ số

Reason-phrase = < TEXT-UTF8, excluding CR, LF >

Status-code gồm 3 chữ số. Chữ số đầu tiên định nghĩa loại đáp ứng, hai số sau không có vai trò phân loại. Bản SIP 2.0 định nghĩa 6 giá trị của số đầu tiên:

-1xx: Informational:Nhận và xử lý yêu cầu.

- 2xx: Success:Thành công.

- 3xx: Redirection: Chuyển tiếp yêu cầu.

- 4xx: Client-Error:Lỗi người dùng, cú pháp của yêu cầu lỗi.

- 5xx: Server-Error:Lỗi máy chủ.

- 6xx: Global-Failure:Lỗi chung (yêu cầu không được đáp ứng tại mọi Server).

Mã đáp ứng SIP có thể mở rộng được. Các ứng dụng SIP không yêu cầu phải hiểu rõ về ý nghĩa của tất cả mã đáp ứng được đăng ký mà chỉ cần hiểu các loại mã đáp ứng, ý nghĩa của chữ số đầu tiên.

Tiêu đề bản tin Response

Tương tự như tiêu đề của bản tin Request, tiêu đề bản tin Response cũng được chia thành 3 dạng: General-header, Response-header và Entity-header

General-header=Acept

| Acept-Encoding

| Acept-Language

| Call-ID

| Contact

| CSeq

| Date

| Encryption

| Expires

| From

| Record-route

| Timestamp

| To

| Via

Response-header = Allow

| Proxy-Authenticate

| Retry-After

| Server

| Unsupported

| Warning

| www-Authenticate

Entity-header= Content-Encoding

| Content-Length

| Content-Type

5.4.3 Trường tiêu đề trong bản tin SIP

Các trường tiêu đề của SIP giống như các trường tiêu đề của HTTP về cả cú pháp và ngữ nghĩa. Các trường tiêu đề khác có thể được bổ sung khi có yêu cầu. Proxy không loại bỏ hoặc thay đổi các trường tiêu đề mà không được định nghĩa trong đặc tả này.

1) Trường tiêu đề chung (general-header)

Các trường tiêu đề chung áp dụng cho tất cả các bản tin yêu cầu và bản tin đáp ứng. Các tên trường tiêu đề chung có thể được mở rộng khi kết hợp với sự thay đổi về phiên bản giao thức. Tuy nhiên, trường tiêu đề mới hoặc đang thử nghiệm có thể có nghĩa như các trường tiêu đề chung nếu tất cả các bên trong phiên truyền thông đều nhận biết được chúng là các trường tiêu đề chung. Các trường tiêu đề chung không nhận biết được xử lý như các trường tiêu đề thực thể (entity-header).

2) Trường tiêu đề thực thể (entity-header)

Các trường tiêu đề thực thể định nghĩa siêu thông tin về thân bản tin hoặc nếu thân bản tin không có mặt thì nó thể hiện các tài nguyên để định danh theo yêu cầu. Thuật ngữ "entity-header" là thuật ngữ của HTTP/1.1, ở đó thân của đáp ứng có thể chứa phiên bản thay đổi của thân bản tin. Thân của bản tin ban đầu được gọi là "thực thể" (entity).

3) Trường tiêu đề yêu cầu (request-header)

Các trường tiêu đề yêu cầu cho phép Client chuyển qua các thông tin bổ sung cho yêu cầu và các thông tin về Client đến Server. Các trường này hoạt động như các bộ thay đổi theo yêu cầu với nghĩa tượng trưng như các tham số của ngôn ngữ lập trình.

4) Trường tiêu đề đáp ứng (response-header)

Các trường tiêu đề đáp ứng cho phép Server bỏ qua các thông tin bổ sung về đáp ứng khi không thể đặt trong trạng thái đường dây (status-line). Các trường tiêu đề này đưa ra thông tin về Server và về các truy nhập xa hơn đến tài nguyên được định danh bởi Request-URI.

Tuy nhiên, trường tiêu đề mới hoặc đang thử nghiệm có thể có nghĩa như trường "Response-header" nếu tất cả các bên trong phiên truyền thông đều nhận biết được chúng là các trường "Response-header". Các trường tiêu đề không nhận biết được xử lý như các trường tiêu đề thực thể.

5) Các tiêu đề end-to-end và hop-by-hop

Các tiêu đề end-to-end phải được phát đi mà không có sự khác nhau nào qua tất cả các Proxy, trong khi tiêu đề hop-by-hop có thể thay đổi hoặc bổ sung tại các Proxy.

6) Khuôn dạng trường tiêu đề (header-field-format)

Các trường tiêu đề kể trên (general-header, request-header, response-header, và entity-header) đều có khuôn dạng giống nhau. Chúng bao gồm tên trường, theo sau bởi dấu ":" và giá trị của trường. Khuôn dạng của trường tiêu đề bản tin:

message-header = field-name ":" [field-value]CRLF

field-name = token

field-value = *(field-content / LWS)

field-content = < Là các Octet cấu thành field-value và hoặc là *TEXT-UTF8 hoặc là các tổ hợp token, separators và quoted-string >

7) Tiêu đề chấp nhận (Accept)

Khi không chấp nhận tiêu đề hiện có tức là có sự xuất hiện của trường Accept, Server sẽ phát đi một giá trị mặc định của ứng dụng sdp. Ví dụ:

accept: application/sdp; lever = 1, application/x-private, text/html

8) Tiêu đề Accept-encoding

Trường tiêu đề yêu cầu Accept-encoding cũng giống như tiêu đề Accept và biểu diễn sự chấp nhận của đáp ứng.

9) Tiêu đề Accept-language

Cú pháp phần tiêu đề ngôn ngữ chấp nhận quy định về lệnh trong ngôn ngữ dựa vào tham số áp dụng vào SIP tốt như thế nào. Trường tiêu đề yêu cầu Accept-language cho phép Client thông báo cho Server ngôn ngữ nó mong muốn để nhận reason-phrases, miêu tả phiên hay mã trạng thái được truyền tải bởi thân bản tin. Một Proxy có thể sử dụng trường này để trợ giúp việc lựa chọn đích đối với các cuộc gọi.

10) Tiêu đề cho phép (Allow)

Trường tiêu đề thực thể (entity-header) liệt kê tập hợp các chỉ thị được hỗ trợ bởi tài nguyên định nghĩa bởi yêu cầu URI. Mục đích của trường này là thông báo đến người nhận các chỉ thị giá trị kết hợp với tài nguyên một cách chính xác. Trường tiêu đề Allow có mặt trong đáp ứng 405 và trong đáp ứng Options.

Allow = "Allow" ":" 1#method

11) Tiêu đề cho phép (Authorization)

UAS muốn xác định bản thân mình với máy chủ sau khi nhận được đáp ứng 401, nó có thể làm điều này bằng cách đưa trường tiêu đề yêu cầu Authorization vào một yêu cầu. Giá trị trườngAuthorization bao gồm các ủy nhiệm chứa thông tin cho phép UASbiết phạm vi hoạt động của tài nguyên đã được yêu cầu.

12) Tiêu đề Call-ID (chỉ số cuộc gọi)

Trường tiêu đề chung Call-ID xác định một lời mời thành viên hoặc tất cả các đăng ký của một Client thành viên một cách duy nhất. Chú ý rằng một hội nghị đa

phương tiện đơn giản có thể thêm vài cuộc gọi với các Call-ID khác nhau, nếu người sử dụng mời nhiều lần khác nhau đến cùng một hội nghị.

Đối với một yêu cầu INVITE, một UAS bị gọi không nên cảnh báo với người sử dụng đó nếu người sử dụng đã đáp ứng lại Call-ID trong yêu cầu INVITE. Nếu người sử dụng đã là một thành viên của hội nghị và các tham số của hội nghị đã được chứa trong một miêu tả phiên không có thay đổi thì UAS bị gọi này có thể chấp nhận cuộc gọi bất chấp Call-ID. Một lời mời cho Call-ID hoặc phiên hiện tại có thể thay đổi các tham số của hội nghị. Một ứng dụng của Client có thể đưa ra một chỉ thị đơn giản tới người sử dụng về các tham số của hội nghị đã được thay đổi và chấp nhận cuộc gọi tự động hoặc nó có thể yêu cầu người sử dụng xác nhận.

Một người sử dụng có thể được mời đến cùng một hội nghị hoặc cuộc gọi sử dụng nhiều Call-ID khác nhau. Client có thể loại phát hiện những bản sao này bằng cách nó sử dụng các nhận dạng trong miêu tả phiên.

Chỉ thị Register (đăng kí) và Option (tùy chọn) sử dụng giá trị Call-ID để yêu cầu được giống hoàn toàn với đáp ứng. Tất cả các yêu cầu Register đưa ra bởi một Client phải có cùng một Call-ID.

Khuôn dạng:

Call-ID = (“Call-ID”|“i”)“:”local-id“@”host

Local-id = 1*uric

13) Tiêu đề giao dịch (Contact)

Trường tiêu đề chung Contact có thể xác nhận trong yêu cầu INVITE, ACK, Register và trong đáp ứng 1xx, 2xx, 3xx và 485. Nó cung cấp URL để người sử dụng có thể có các giao tiếp xa hơn. Các yêu cầu INVITE và ACK có thể chứa tiêu đề Contact để chỉ ra vị trí khởi tạo yêu cầu.

Đáp ứng INVITE 1xx: Một UAS gửi một đáp ứng 1xx có thể chèn vào tiêu đề đáp ứng Contact. Nó cũng có ý nghĩa như đáp ứng 1xx và đáp ứng INVITE 2xx. Chú ý rằng yêu cầu CANCEL không được gửi đến địa chỉ đó.

Đáp ứng INVITE 2xx: Một Server gửi một đáp ứng 2xx có thể chèn vào một tiêu đề đáp ứng Contact để chỉ ra vị trí SIP trong cùng một Call-ID. Trường tiêu đề Contact chứa địa chỉ của Server hoặc Proxy, nếu trạm ở phía sau tường lửa. Giá trị tiêu đề được sao chép vào trường Request-URI của yêu cầu tiếp theo cho cuộc gọi này. Nếu đáp ứng không chứa tiêu đề Record-route thì địa chỉ trong trường tiêu đề Contact được bổ sung như phần tử cuối cùng trong trường tiêu đề route.

Yêu cầu Register: Yêu cầu Register có thể chứa trường tiêu đề Contact, nó chỉ ra vị trí người sử dụng có thể đến. Yêu cầu Register định nghĩa trường "*" chỉ được sử dụng với expire "0" để loại bỏ tất cả các đăng ký người sử dụng đặc biệt. Một tham số expire tùy chọn chỉ ra thời gian kết thúc của đăng ký. Nếu Contact nào không có tham số expire thì trường tiêu đề expire được sử dụng như giá trị mặc định.

Đáp ứng 2xx Register: Một đáp ứng Register có thể quay về tất cả các vị trí mà người sử dụng hiện tại có thể đạt đến. Một tham số expire mặc định chỉ ra thời gian kết thúc của đăng ký.

Đáp ứng 3xx và 485: Trường tiêu đề đáp ứng Contact có thể sử dụng chung với mã đáp ứng 3xx và 485 để chỉ ra một hoặc nhiều địa chỉ thay thế. Nó có thể xuất hiện trong các chỉ thị BYE, INVITE, OPTION. Trường tiêu đề Contact chứa URI đưa ra vị trí mới hoặc tên của User để thử hoặc có thể là các tham số truyền bổ sung đặc biệt. Đáp ứng 300, 301, 302, 485 chứa trường Contact với địa chỉ URI mới. Client sao chép các phần tử "user", "password", "host", "port" và "user param" của URIContact vào URI yêu cầu định hướng yêu cầu đến địa chỉ đặc tả bởi tham số "maddr" và "port", sử dụng giao thức truyền tải trong "transport param". Một trường tiêu đề đáp ứng Contact có thể chứa bất kỳ URI thích hợp để chỉ ra bị gọi nào có thể đạt được, không hạn chế đến SIP-URL.

Các tham số sau được định nghĩa:

- q: giá trị q chỉ ra ưu tiên tương đối giữa vị trí cho trước (0<q<1), q càng lớn độ ưu tiên càng cao.

- Action: tham số này được sử dụng khi đăng ký với yêu cầu Register. Nếu tham số này không được đặc tả thì hoạt động xảy ra phụ thuộc vào cấu hình Server, khi đó trong các đáp ứng của nó, các bản sao nên chỉ ra chế độ được sử dụng. Tham số này không được chú ý trong các yêu cầu khác nhau.

- Expire: tham số expire chỉ ra thời gian hiệu lực của URI.

14) Tiêu đề mã hóa nội dung (Content-Encoding)

Giá trị của trường tiêu đề thực thể Content-Encoding chỉ ra các mã hóa phù hợp được ứng dụng trong phần thân bản tin, vì vậy cơ chế giải mã phải được ứng dụng để đạt được kiểu truyền thông trong trường tiêu đề Content-Type. Nếu nhiều mã hóa được dùng trong bản tin thì mã hóa phải được sắp xếp phù hợp theo thứ tự chúng được dùng.

Client có thể dùng mã hóa Content và phần thân của yêu cầu. Nếu Server không có khả năng mã hóa phần thân hoặc phần không thể nhận biết được bất kỳ giá trị

Content-Encoding nào, thì nó phải gửi đáp ứng 415, liệt kê các mã hóa có thể chấp nhận trong tiêu đề Accept-Encoding. Một Server có thể áp dụng Content-Encoding vào phần chính trong các đáp ứng. Server chỉ được sử dụng các mã hóa đã liệt kê trong tiêu đề Accept-Encoding trong yêu cầu.

15) Tiêu đề độ dài nội dung (Content-Length)

Giá trị trường tiêu đề Content-Length chỉ ra kích cỡ phần chính của bản tin. Là số Octet tính theo thập phân gửi đến phía nhận. Các ứng dụng nên dùng trường này để chỉ ra kích cỡ phần chính của bản tin đã truyền đi, bất kể kiểu phương tiện gì của thực thể. Giá trị Content-Length ≥ 0 đều được sử dụng (giá trị bằng không khi phần thân trong bản tin không có). Nếu Server nhận yêu cầu UDP có trường Content-Length, mà giá trị của nó lớn hơn kích cỡ của phần thân bản tin gửi trong yêu cầu, thì Client tạo ra đáp ứng 400. Nếu có dữ liệu bổ sung trong gói UDP sau khi byte cuối cùng của thân bản tin được đọc, thì Server phải xử lý phần dữ liệu như bản tin riêng. Điều này cho phép nhiều bản tin được đặt trong cùng một gói UDP.

16) Tiêu đề kiểu nội dung (Content-Type)

Giá trị trường tiêu đề Content-Type chỉ ra kiểu truyền thông của thân bản tin gửi đến phía nhận.

17) Tiêu đề CSeq

CSeq do chính Client bổ sung vào mỗi yêu cầu. Trường tiêu đề CSeq trong thân bản tin yêu cầu chứa chỉ thị yêu cầu và số lượng chuỗi được chọn theo yêu cầu của Client, với một giá trị Call-ID. Số lượng chuỗi phải là một số nguyên không dấu 32 bit. Giá trị khởi tạo của nó là tùy ý nhưng phải nhỏ hơn 2^{31}. Các yêu cầu ACK và CANCEL phải chứa cùng giá trị CSeq như trong yêu cầu INVITE trong khi yêu cầu BYE với CSeq không cao hơn sẽ tạo ra đáp ứng 400.

Một Server của UAS phải nhờ số lượng chuỗi cao nhất đối với bất kỳ yêu cầu INVITE nào trong cùng giá trị với Call-ID, Server phải đáp lại bất kỳ yêu cầu INVITE nào với cùng một số lượng chuỗi thấp hơn.

18) Tiêu đề thời gian (Date)

Là một trường tiêu đề chung phản ánh thời gian khi yêu cầu hoặc đáp ứng được gửi đi lần đầu. Vì vậy, các bản tin truyền đạt có cùng giá trị trường tiêu đề dữ liệu như bản tin đầu tiên.

19) Tiêu đề mã hóa (Encryption)

Trường tiêu đề mã hóa chung đặc tả nội dung được mã hóa. Trường tiêu đề này thường sử dụng cho các yêu cầu và các đáp ứng end-to-end (đầu cuối tới đầu cuối). Yêu cầu được mã hóa dựa trên khóa công khai của thực thể gọi là trường tiêu đề "To".

Đáp ứng được mã hóa dựa trên khóa công khai truyền trong trường tiêu đề khóa đáp ứng. Chú ý rằng bản thân khóa công khai không được sử dụng để mã hóa. Điều này phụ thuộc vào từng thuật toán được sử dụng.

Đối với bất kỳ bản tin đã được mã hóa nào, thì ít nhất có phần thân bản tin và có thể các trường tiêu đề khác được mã hóa để giải mã thân bản tin, sau đó thực hiện ghép bản tin ban đầu đến đường dây yêu cầu và đến các tiêu đề của bản tin ban đầu. Tiêu đề bản tin trong phần đã giải mã thay thế hoàn toàn tiêu đề bản tin trong phần bản tin ban đầu với các trường có cùng tên.

Mã hóa chỉ để cung cấp sự bí mật: phía đích không đảm bảo rằng yêu cầu hoặc đáp ứng từ phía gửi được liệt kê trong tiêu đề bản tin From mà phía đích chỉ đảm bảo rằng phía gửi sử dụng khóa công khai của phía đích. Tuy nhiên, các Proxy không có khả năng thay đổi yêu cầu hay đáp ứng.

Vì các Proxy có thể dựa vào sự quy định trước của chúng để kết hợp bất kỳ trường tiêu đề SIP nào nên không có sự đảm bảo rằng các yêu cầu đã mã hóacó các trường tiêu đề "ẩn" (hide) sẽ cùng đến đích như các yêu cầu không mã hóa được định danh khác nhau.

20) Tiêu đề thực thể (Expires)

Trường tiêu đề thực thể expire đưa ra thời gian và ngày mà nội dung bản tin sẽ hết hiệu lực. Trường này chỉ được định nghĩa trong chỉ thị Register và INVITE.

Đối với Register: Nó là trường tiêu đề đáp ứng và yêu cầu trong bản tin Register, Client chỉ ra đăng ký giá trị trong bao lâu. Trong đáp ứng Server chỉ ra thời gian hết hiệu lực sớm nhất của các đăng ký. Server có thể lựa chọn khoảng thời gian ngắn hơn trong yêu cầu từ Client nhưng không nên lâu hơn.

Đối với INVITE: Nó là trường tiêu đề đáp ứng và yêu cầu. Trong một yêu cầu, chủ gọi có thể hạn chế giá trị của yêu cầu, ví dụ Client có thể hạn chế khoảng thời gian tìm kiếm một yêu cầu hội nghị.

21) Tiêu đề xuất phát (From)

Các đáp ứng và yêu cầu phải có trường tiêu đề chung From để chỉ ra bộ khởi đầu yêu cầu. Server sao chép trường tiêu đề From từ yêu cầu vào đáp ứng. Trường From có

thể có tham số "Tag". Khi cần thiết nó có thể trình bày 2 trường đặc biệt của người sử dụng chia sẻ một địa chỉ SIP có thể tạo các yêu cầu gọi ứng với cùng một Call-ID. Giá trị "Tag" trong suốt một cuộc gọi đã định danh bằng Call-ID.

22) Tiêu đề ẩn (Hide)

Client sử dụng trường tiêu đề yêu cầu "hide"để chỉ ra rằng nó muốn tuyến bao gồm tiêu đề Via được ẩn khi qua các Proxy và Agent người sử dụng. Nó có thể có 2 dạng: Route Hide và Hop Hide.

Trường tiêu đề Hide có thể được bổ sung bởi Agent sử dụng Client, cũng có thể bởi Proxy dọc theo các tuyến đường. Nếu một yêu cầu chứa trường tiêu đề Route Hide thì tất cả các Proxy phía sau nên ẩn các chặng trước đó của chúng. Nếu một yêu cầu chứa "Hide-hop" thì chỉ Proxy tiếp theo sẽ ẩn hop trước đó và sau đó loại bỏ tùy chọn Hide trừ khi nó muốn duy trì tên ẩn.

Một Server ẩn chặng trước đó bằng cách mã hóa phần "host" và "sport" của hầu hết các tiêu đề "Via" đỉnh theo thuật toán nó chọn. Server nên bổ sung thêm "salt" đến "host" và "sport" ưu tiên thông tin cho mã hóa để ngăn chặn các Proxy tiếp theo có nguy hiểm từ chỉ dẫn phần trước của tuyến dựa vào sự xem xét tiêu đề "Via" đã mã hóa. Một Server có khả năng dấu các tiêu đề "Via" phải cố gắng giải mã tất cả các tiêu đề "Via" có chú ý là "hide" để thực hiện các phát hiện vòng.

Một trạm không phải bổ sung trường tiêu đề "Hide-hop" trừ khi nó đảm bảo rằng nó sẽ gửi yêu cầu cho đích này đến cùng "hop" tiếp theo. Một Client yêu cầu "Hide-route" chỉ tin vào việc duy trì đường yêu cầu riêng nếu nó được gửi yêu cầu đến Proxy an toàn.

23) Tiêu đề Max-Forward

Trường tiêu đề yêu cầu Max-Forward sử dụng chỉ thị SIP để hạn chế số Proxy hoặc số cổng trên hướng mà yêu cầu đi đến Server tiếp theo. Giá trị Max-Forward là số nguyên thập phân chỉ ra số lần duy trì bản tin yêu cầu này được cho phép theo hướng truyền. Mỗi Proxy nhận hoặc cổng nhận yêu cầu chứa trường tiêu đề Max-Forward phải kiểm tra và cập nhật giá trị của nó ưu tiên về việc định hướng yêu cầu:

- Nếu giá trị là 0: phía nhận không phải định hướng trước yêu cầu. Thay vào đó, đối với chỉ thị Option và Register, nó phải đáp lại như một phía nhận cuối cùng, còn đối với các chỉ thị khác thì nó trở về trạng thái 483.

- Nếu giá trị là lớn hơn 0: bản tin đã định hướng phải chứa trường Max-Forward đã cập nhật với giá trị giảm đi 1.

24) Tiêu đề Organization

Trường tiêu đề chung Organization đưa ra tên của nơi mà thực thể đưa ra yêu cầu hoặc đáp ứng. Trường này có thể được sử dụng bằng phần mềm Client để lọc các cuộc gọi.

25) Tiêu đề ưu tiên (Priority)

Trường tiêu đề ưu tiên (Priority) chỉ ra độ ưu tiên của yêu cầu khi Client nhận được.

26) Tiêu đề Proxy-Authenticate

Trường tiêu đề yêu tiên Proxy-Authenticate phải được kể đến như một phần của đáp ứng 407. Giá trị của trường bao gồm các nhận dạng để chỉ ra cấu trúc xác thực và các tham số ứng dụng vào Proxy đối với Request-URI này.

27) Tiêu đề (Proxy-Authorization)

Trường tiêu đề yêu cầu Proxy-Authorization cho phép xác thực bản thân nó cho Proxy khi có yêu cầu xác thực. Giá trị trường Proxy-Authorization bao gồm các ủy quyền có chứa thông tin xác thực của Agent người sử dụng về phạm vi hoạt động của tài nguyên được yêu cầu và Proxy.

28) Tiêu đề Proxy-Require

Trường tiêu đề Proxy-Require sử dụng để chỉ ra các điểm nhạy cảm của Proxy được hỗ trợ bởi Proxy.

29) Tiêu đề Record-route

Trường tiêu đề Record-route được bổ sung vào một yêu cầu bởi một Proxy bất kỳ trong đường các yêu cầu tiếp theo cho cùng một nhánh Call. Nó chứa các Request-URI có thể đến để xác định Server Proxy. Mỗi Server Proxy bổ sung Request-URI vào đầu của danh sách. Server sao chép trường tiêu đề Record-route vào trường tiêu đề Route của các yêu cầu tiếp theo trong cùng một nhánh Call với thứ tự các yêu cầu ngược lại để yêu cầu là gần nhất với UAC.

Nếu trường tiêu đề chứa trường tiêu đề Contact thì UAC chủ gọi bổ sung nội dung của nó như một tiểu đề Route cuối cùng. Trừ khi điều này gây lặp, bất kỳ Client nào đều phải gửi yêu cầu tiếp theo bất kỳ đối với nhánh gọi này đến Request-URI đầu tiên trong trường tiêu đề yêu cầu Route và loại bỏ lối vào này.

30) Tiêu đề yêu cầu (Require)

Client sử dụng trường tiêu đề yêu cầu Require để thông báo cho các Server Agent User về các tùy chọn mà Client mong đợi các Server cung cấp để xử lý thích hợp yêu

cầu. Nếu một Server không hiểu tùy chọn thì nó phải đáp lại bằng cách quay trở về mã trạng thái 420 và liệt kê những tùy chọn nó không hiểu trong tiêu đề Unsupported.

31) Tiêu đề Response-Key

Client có thể sử dụng trường tiêu đề Response-Key để yêu cầu khóa mà Agent User sử dụng mã hóa đáp ứng đó. Key-stem đưa ra kiểu mã hóa được sử dụng cho đáp ứng.

32) Tiêu đề Retry-after

Trường tiêu đề chung Retry-after có thể được sử dụng với đáp ứng 563 để chỉ ra dịch vụ mà Client yêu cầu không có giá trị và với đáp ứng 404; 600 để chỉ ra khi nào phía bị gọi lại khả dụng lại. Giá trị của trường này có thể là SIP-date hoặc số nguyên theo sau thời gian đáp ứng.

33) Tiêu đề Route

Trường tiêu đề Route xác định tuyến diễn ra yêu cầu. Mỗi trạm loại bỏ lối vào đầu tiên và sau đó các Proxy yêu cầu trạm liệt kê lối vào đó và cũng sử dụng nó như Request-URI.

34) Tiêu đề Server

Trường tiêu đề đáp ứng Server chứa thông tin về phần mềm được sử dụng bởi User Agent Server để xử lý yêu cầu.

35) Tiêu đề Subject

Tiêu đề này được sử dụng để cung cấp các tổng kết hoặc chỉ ra bản chất của Call, cho phép lọc Call mà không có sự phân tích các miêu tả phiên.

36) Tiêu đề Timestamp

Trường tiêu đề chung Timestamp thông báo khi nào Client gửi yêu cầu đến Server. Giá trị của Timestamp chỉ ảnh hưởng đến Client và nó có thể sử dụng bất kỳ đơn vị thời gian nào. Server phải hồi âm với cùng giá trị chính xác và có thể bổ sung thêm số dấu phẩy động để chỉ ra số giây đã qua từ khi nó nhận được yêu cầu.

37) Tiêu đề To

Trường tiêu đề chung To đặc tả phía nhận của yêu cầu, có cùng cú pháp SIP-URI với trường "From". Các đáp ứng và yêu cầu phải thừa nhận trường tiêu đề chung "To", chỉ ra phía nhận trong yêu cầu. Yêu cầu và đáp ứng phải có chung trường tiêu đề "To". UAS hoặc Redirect Server sẽ sao chép trường "To" vào trong đáp ứng và bổ sung thêm

thông số "Tag" nếu yêu cầu chứa quá một trường "Via". Tham số "Tag" trong tiêu đề To dùng để phân biệt đáp ứng 400 (yêu cầu lỗi) nếu nó nhận được một trường tiêu đề To có cấu trúc mà nó không hiểu được.

38) Tiêu đề Unsupported

Trường tiêu đề đáp ứng Unsupported (không được hỗ trợ) liệt kê các đặc điểm mà Server không hỗ trợ trong phiên truyền thông.

39) Tiêu đề User-Agent

Trường tiêu đề chung User-Agent chứa thông tin về User Agent Client khởi tạo bản tin.

40) Tiêu đề Via

Trường tiêu đề Via chỉ ra tuyến mà yêu cầu đi qua do đó ngăn chặn được các yêu cầu bị lặp và đảm bảo trả lời trên cùng đường yêu cầu. Sau đây ta xét cụ thể đối với các trường tiêu đề Via:

a) Trường tiêu đề Via Request

Client khởi tạo yêu cầu phải chèn vào yêu cầu một trường Via. Một trường Via chứa tên trạm, địa chỉ mạng, số cổng mặc định hoặc số cổng để nhận được đáp ứng. Một Proxy phải kiểm tra trường tiêu đề Via cao nhất để đảm bảo rằng nó chứa địa chỉ mạng chính xác của người gửi. Nếu địa chỉ người gửi là không chính xác, Proxy phải bổ sung thuộc tính "Received".

Một Proxy gửi một yêu cầu đến địa chỉ đa hướng thì nó phải bổ sung tham số "maddr" và tham số "ttl" vào trường tiêu đề Via. Nếu Proxy nhận yêu cầu chứa tham số "maddr" trong trường Via cao nhất thì nó phải gửi đáp ứng đến địa chỉ đa hướng đã liệt kê trong tham số "maddr".

b) Trường tiêu đề Via Receiver-Tagged

Thông thường mỗi trạm gửi hoặc định hướng bản tin SIP phải bổ sung vào trường Via để chỉ ra tuyến ngang qua. Tuy nhiên, có thể NTA thay đổi địa chỉ nguồn và cổng của yêu cầu, trong trường hợp này trường tiêu đề Via không thể tin vào việc trả lời tuyến. Để ngăn chặn điều này, một Proxy nên kiểm tra trường tiêu đề Via cao nhất để đảm bảo rằng nó chứa địa chỉ mạng chính xác của người gửi, khi được nhìn từ phía Proxy. Nếu địa chỉ người gửi không chính xác, Proxy phải bổ sung tham số "received" vào trường tiêu đề Via, trường này được chèn vào bởi Hop trước đó.

c) Trường tiêu đề Via Response

Trường tiêu đề Via trong đáp ứng được xử lý bởi Proxy hoặc UAC theo quy tắc sau:

Trường tiêu đề Via đầu tiên chỉ ra Proxy hoặc Client xử lý đáp ứng này. Nếu không thì bản tin sẽ bị loại bỏ.

Nếu không có trường tiêu đề Via thứ hai thì đáp ứng này được định ra cho Client này. Việc xử lý phụ thuộc vào trường Via có chứa tham số "maddr" hay không, hay nó là một trường receiver-tagged.

+ Nếu trường Via chứa tham số "maddr": Khi đó việc gửi trường đến địa chỉ đa hướng được liệt kê ở đây, cổng sử dụng được chỉ ra trong "sent-by" hoặc cổng 5060 nếu không có cổng nào có mặt. Đáp ứng được gửi sử dụng TTL chỉ ra trong tham số "ttl" hoặc TTL=1 khi tham số này không có mặt.
+ Nếu trường Via không chứa tham số "maddr" và nó là một trường "Receiver-tagged": thì khi đó bản tin sẽ được gửi đến địa chỉ trong tham số "received", sử dụng cổng chỉ ra trong giá trị "sent-by", hoặc sử dụng cổng 5060 nếu tham số này không có mặt.
+ Nếu không rơi vào các trường hợp trên thì bản tin được gửi đến địa chỉ được chỉ ra bởi giá trị "sent-by" trong trường tiêu đề Via thứ hai.

d) User-Agent và Redirect Server

UAS hoặc Server gửi một đáp ứng dựa trên các quy định sau:

+ Nếu trường tiêu đề Via đầu tiên trong yêu cầu chứa tham số "maddr" thì đáp ứng được gửi đến địa chỉ quảng bá được liệt kê ở đó, sử dụng cổng chỉ ra trong "sent-by" hoặc cổng 5060 khi các cổng đó không có mặt.
+ Nếu địa chỉ trong giá trị "sent-by" của trường Via đầu tiên khác với địa chỉ nguồn của gói tin thì gửi đáp ứng đến địa chỉ nguồn của gói hoạt động giống bởi trường hợp xử lý trường tiêu đề Via Receiver-tagged.
+ Nếu các điều kiện trên không đúng thì đáp ứng được gửi đến địa chỉ chứa giá trị "sent-by". Nếu đáp ứng được gửi sử dụng TCP, sử dụng kết nối TCP đang tồn tại nếu nó khả dụng.

e) Syntax

Các mặc định cho "Protocol-name" (tên giao thức) và "transport" (giao thức truyền) là "SIP" và "UDP". Tham số "maddr" biểu diễn địa chỉ đa hướng và tham số "ttl" mô tả giá trị TTL (Time To Live) chỉ được kể đến nếu yêu cầu được gửi qua địa chỉ đa hướng. Tham số "received" được bổ sung chỉ trong trường Via Receiver-added.

Đối với lý do riêng, Client hoặc Proxy có thể muốn dấu thông tin Via của nó bằng việc mã hóa nó. Tham số "hide" được đưa vào nếu trường tiêu đề được dấu bởi Proxy ngược. Tham số "branch" được thiết kế đến bởi mỗi Proxy nhánh. Dấu hiệu phải là đơn nhất đối với mỗi yêu cầu phân biệt khi Proxy đánh dấu.

Các yêu cầu CANCEL phải có cùng giá trị "branch" nhánh tương ứng với yêu cầu đã đánh dấu. Khi đáp ứng đến một Proxy nó có thể sử dụng giá trị branch để chỉ ra nhánh tương ứng với đáp ứng. Một Proxy tạo ra một yêu cầu đơn (không dấu) ứng với đáp ứng. Một Proxy tạo ra một yêu cầu đơn có thể chèn tham số "branch". Định danh này có giá trị đơn nhất chỉ với một tập các yêu cầu đẳng cấp.

41) Tiêu đề Warning

Trường tiêu đề đáp ứng Warning được sử dụng để mang các thông tin bổ sung về trạng thái của đáp ứng. Một Server bất kỳ có thể bổ sung tiêu đề Warning vào đáp ứng, Server Proxy phải đặt tiêu đề Warning bổ sung vào trước tiêu đề nhận thức bất kỳ.

5.4.4 Mã đáp ứng của bản tin SIP

Mã đáp ứng (Response-Code) của SIP được mở rộng từ mã đáp ứng trong HTTP/1.1, chỉ có những mã phù hợp của HTTP/1.1 mới được đưa vào trong SIP.

Mã đáp ứng 1xx

Đáp ứng Informational cho biết Server hoặc Proxy liên lạc đang thực hiện các hoạt động và vẫn chưa nhận được một đáp ứng cuối cùng. Client phải đợi một đáp ứng tiếp theo từ Server. Server gửi một đáp ứng 1xx nếu như nó muốn đợi 200 ms để nhận được một đáp ứng cuối cùng. Một Server có thể phát ra nhiều đáp ứng 1xx. Chú ý rằng đáp ứng 1xx không có độ tin cậy truyền dẫn, có nghĩa là nó không phải là nguyên nhân khiến cho Client gửi một ACK. Server có thể tự do phát lại các đáp ứng Informational và Client có thể kiểm tra trạng thái hiện tại của việc xử lý gọi bằng cách gọi lại các yêu cầu. Sau đây ta xét một số đáp ứng 1xx cơ bản:

1) Đáp ứng 100 - Trying (thử)

Một số hoạt động không xác định sẽ được đại diện cho cuộc gọi nhưng User không được định vị.

2) Đáp ứng 180 - Ringing (báo chuông)

UAS vừa định vị mộtvị trí có thể, nơi mà đầu cuối được đăng ký gần đây nhất và UAS cố gắng để báo cho User.

3) Đáp ứng 181 - Call is being forwarded (cuộc gọi sắp được chuyển tới)

Một Proxy Server dùng mã trạng thái để thông báo là cuộc gọi sắp tới đích.

4) Đáp ứng 182 - Queued (xếp hàng)

Phía bị gọi tạm thời không sẵn sàng, phía chủ gọi sẽ quyết định xếp hàng các cuộc gọi thay cho việc bỏ chúng đi. Khi chủ gọi trở nên sẵn sàng nó sẽ trả về một đáp ứng trạng thái cuối cùng thích hợp. Server có thể phát ra một số đáp ứng 182 để cập nhật phía chủ gọi và trạng thái của hàng đợi gọi.

Mã đáp ứng 2xx

Yêu cầu đã được thực hiện thành công và kết thúc một tìm kiếm. Một đáp ứng tiêu biểu của 2xx đó là đáp ứng 200 OK.

Yêu cầu đã được thực hiện thành công. Thông tin được trả lại với đáp ứng tùy thuộc vào chỉ thị dùng trong yêu cầu. Ví dụ:

BYE: Cuộc gọi đã kết thúc, thân bản tin rỗng.

CANCEL: Tìm kiếm bị hủy bỏ, thân bản tin rỗng.

INVITE: User được gọi đồng ý tham gia hội thoại, thân bản tin cho biết khả năng của User được gọi.

OPTION: Người gọi đồng ý chia sẻ khả năng.

REGISTER: Việc đăng ký đã thành công.

Mã đáp ứng 3xx

Các đáp ứng 3xx đưa ra thông tin về vị trí mới của User hoặc về dịch vụ để hoàn thành cuộc gọi. Chúng có thể kết thúc tìm kiếm hiện tại và có thể tạo ra một tìm kiếm mới nếu nó phù hợp. Các đáp ứng 3xx không yêu cầu địa chỉ trong trường Via yêu cầu trong trường tiêu đề "Contact". Để tránh lặp với các địa chỉ trước, UAC hay Proxy phải kiểm tra xem có địa chỉ trả lại nào của Redirect Server giống với một địa chỉ trước đó không. Một số đáp ứng Redirect 3xx cơ bản:

1) Đáp ứng 300-Multiple Choices (nhiều sự lựa chọn)

Địa chỉ trong yêu cầu có thể có vài lựa chọn ứng với các vị trí cụ thể của nó và User hay UA có thể lựa chọn một địa chỉ ưu tiên trong đó để gửi lại yêu cầu. Đáp ứng gồm một thực thể chứa danh sách về đặc điểm nguồn và vị trí mà từ đó User hay UA có thể lựa chọn một vị trí thích hợp nhất để đăng ký. Khuôn dạng của thực thể được chỉ định bởi kiểu truyền thông trong trường tiêu đề Content-type. Khác với HTTP, đáp ứng

SIP có thể gồm vài Contact hay một danh sách địa chỉ trong trường Contact. UA có thể dùng các giá trị của trường Contact để tự động gửi lại hoặc yêu cầu User xác nhận một sự lựa chọn.

2) Đáp ứng 301-Moved Permanently (dịch chuyển cố định)

User không dựa trên địa chỉ trong trường Request-URI và yêu cầu Client để thử lại một địa chỉ mới trong trường tiêu đề Contact. Chủ gọi sẽ cập nhật danh sách cục bộ, danh sách địa chỉ và vị trí của User rồi gửi lại các yêu cầu đến danh sách địa chỉ.

3) Đáp ứng 302-Moved Temporarily (dịch chuyển tạm thời)

Yêu cầu Client thử lại yêu cầu tại một địa chỉ mới trong trường Contact. Khoảng thời gian phát lại yêu cầu được trình bày trong tiêu đề expire. Nếu như không có thời gian kết thúc rõ ràng thì địa chỉ đó chỉ sử dụng cho cuộc gọi hiện tại và không được dùng cho các cuộc gọi tiếp theo.

4) Đáp ứng 305-User Proxy (ủy quyền User)

Yêu cầu truy nhập tài nguyên thông qua Proxy. Trường Contact chứa URI của Proxy. Đáp ứng này chỉ được phát ra bởi UAS.

5) Đáp ứng 380-Alternative Server (luân chuyển Server)

Cuộc gọi chưa hoàn thành nhưng dịch vụ lựa chọn (thay đổi) được sử dụng. Dịch vụ lựa chọn được miêu tả trong thân bản tin của đáp ứng.

Mã đáp ứng 4xx

Đáp ứng 4xx chỉ rõ các đáp ứng lỗi từ một Server riêng biệt. Client không thể gửi lại cùng một yêu cầu mà không có sự thay đổi. Tuy nhiên, yêu cầu giống nhau từ một Server khác thì có thể vẫn thành công. Một số đáp ứng 4xx cơ bản:

1) Đáp ứng 400-Bad Request (yêu cầu lỗi)

Yêu cầu không thể hiểu do sai về cú pháp.

2) Đáp ứng 401-Unauthoried (Không được phép)

Yêu cầu đòi hỏi xác nhận của User.

3) Đáp ứng 403-Forbidden (Cấm thực hiện)

Server hiểu yêu cầu nhưng từ chối thực hiện nó. Yêu cầu không được lặp lại khi có đáp ứng này.

4) Đáp ứng 404-Not Found (không tìm thấy)

Server có thông tin là User không có mặt trong chỉ định vùng trọng trường Request-URI. Trạng thái này được trả lại khi vùng trọng trường Request-URI không khớp với vùng nhận được bởi yêu cầu.

5) Đáp ứng 405-Method Not Allowed (chỉ thị không cho phép)

Chỉ thị trong Request-Line không cho phép nhận biết địa chỉ thông qua trường Request-URI. Đáp ứng này gồm một trường tiêu đề Allow chứa một danh sách các chỉ thị có hiệu lực cho việc nhận biết các địa chỉ.

6) Đáp ứng 480-Temporarily Unavailble (tạm thời không sẵn sàng)

Hệ thống đầu cuối bị gọi đã liên lạc thành công nhưng phía bị gọi lại không sẵn sàng. Đáp ứng này chỉ ra một thời gian tốt hơn để cho cuộc gọi trong tiêu đề Retry-after. Reason Phrase có thể chỉ ra các nguyên nhân chính xác tại sao phía bị gọi lại không sẵn sàng. Giá trị này có thể được thiết lập bởi UA. Đáp ứng này được trả lại bởi một Redirect Server nhận diện được User thông qua Redirect-URI nhưng không có được vị trí hợp lý hiện nay của User đó.

7) Đáp ứng 481-Call Leg/Transaction does not exist

Đáp ứng này được trả lại khi có hai điền kiện sau:

- Server nhận được một yêu cầu BYE mà không phù hợp với các cuộc gọi nhánh "Call-Leg" hiện tại.

- Server nhận được một yêu cầu CANCEL mà không phù hợp với các giao dịch hiện tại.

8) Đáp ứng 484-Address Incomplete (địa chỉ không đầy đủ)

Server nhận được một yêu cầu với địa chỉ đến To hoặc Request-URI mà không đầy đủ. Mã trạng thái này cho phép quay số chồng chéo. Với việc quay số chồng chéo thì Client không thể biết được độ dài của chuỗi số.

9) Đáp ứng 486-Busy Here (đang bận)

Hệ thống đầu cuối bị gọi đã liên lạc thành công nhưng chủ gọi lại không thể tham gia thêm vào cuộc gọi. Đáp ứng này chỉ thị một thời gian tốt hơn cho cuộc gọi trong trường Retry-after.

Mã đáp ứng 5xx

Đáp ứng 5xx là đáp ứng xuất hiện khi chính Server bị lỗi. Sau đây ta xét các đáp ứng lỗi Server hay xảy ra:

1) Đáp ứng 500-Server Internal Error (lỗi nội bộ Server)

Server gặp phải một tình huống bất ngờ ngăn cản nó hoàn thành yêu cầu. Client có thể hiển thị trạng thái lỗi cụ thể và có thể thử lại yêu cầu sau vài giây.

2) Đáp ứng 501- Not Implemented (không thực hiện)

Server không thể hỗ trợ yêu cầu chức năng để hoàn thành yêu cầu. Đáp ứng là một đáp ứng thích hợp khi Server không nhận thức rõ về chỉ thị yêu cầu và khả năng hỗ trợ của nó cho các User.

3) Đáp ứng 502-Bad Gateway (lỗi cổng)

Server khi hoạt động như một Gateway hoặc Proxy, nhận một đáp ứng không hợp lệ từ Downstream Server, nó sẽ cố gắng hoàn thành yêu cầu đó.

4) Đáp ứng 503-Service Unavailble (dịch vụ không sẵn sàng)

Server hiện không có khả năng xử lý yêu cầu do sự quá tải tạm thời hay do bảo dưỡng.

5) Đáp ứng 504-Gateway Time-Out (Cổng quá thời gian)

Server khi hoạt động như một cổng không thể nhận một đáp ứng kịp thời từ Server khác. Nó sẽ truy xuất vào để cố gắng hoàn thành yêu cầu.

6) Đáp ứng 505-Version Not Supported (phiên bản không hỗ trợ)

Server không hỗ trợ cho phiên bản SIP dùng trong bản tin yêu cầu. Đáp ứng này chứa một thực thể miêu tả tại sao không hỗ trợ phiên bản đó và nêu ra các giao thức khác được hỗ trợ bởi Server.

Mã đáp ứng 6xx

Đáp ứng 6xx chỉ ra rằng yêu cầu không được đáp ứng tại mọi Server. Hiện tại hệ thống đang gặp lỗi. Một số đáp ứng 6xx thường xảy ra:

1) Đáp ứng 600-Busy Everywhere (Bận mọi nơi)

Hệ thống đầu cuối bị gọi đã liên lạc thành công nhưng phía bị gọi đang bận và không muốn thực hiện cuộc gọi tại thời điểm này. Đáp ứng này chỉ ra một thời gian

tốt hơn để tiến hành cuộc gọi trong tiêu đề Retry-after. Đáp ứng này chỉ trả lại khi biết rằng không có điểm đầu cuối nào sẽ trả lại đáp ứng tới yêu cầu.

2) Đáp ứng 603-Decline

Máy bị gọi đã liên lạc thành công nhưng User dứt khoát không muốn hoặc không thể tham gia cuộc gọi. Đáp ứng này chỉ ra một thời gian tốt hơn để tiến hành cuộc gọi trong tiêu đề Retry-after.

3) Đáp ứng 604-Does Not exist Anywhere (không tồn tại bất cứ nơi nào)

Server có thông tin chính xác là User được chỉ ra trong trường yêu cầu "To", không tồn tại ở bất cứ nơi nào. Việc tìm kiếm User đó ở một nơi nào đó không mang lại kết quả.

4) Đáp ứng 606-Not Acceptable (không chấp nhận)

UA liên lạc thành công nhưng một số mặt của miêu tả phiên như các yêu cầu về phương tiện, băng thông hay kiểu địa chỉ không được chấp nhận. Đáp ứng 606 có nghĩa là User muốn giao tiếp nhưng không được chấp nhận miêu tả phiên thích hợp. Đáp ứng này gồm một danh sách các nguyên nhân tại sao mô tả phiên không được hỗ trợ chứa trong trường tiêu đề Warning (cảnh báo).

5.5. THÂN BẢN TIN SIP

5.5.1 Thành phần trong thân bản tin

Với bản tin yêu cầu (Request), nó luôn chứa thân bản tin trừ khi có yêu cầu khác. Trong đặc tả này, yêu cầu BYE không chứa một thân bản tin. Còn đối với các chỉ thị ACK, INVITE, OPTION, thân bản tin bao giờ cũng là một miêu tả phiên. Việc sử dụng thân bản tin cho yêu cầu Register (đăng ký) đang được nghiên cứu thêm.

Với bản tin đáp ứng (Response), chỉ thị yêu cầu và mã trạng thái đáp ứng xác định kiểu và sự thể hiện của các thân bản tin. Tất cả các đáp ứng đều gồm một thân bản tin. Thân bản tin cho đáp ứng 1xx chứa thông tin tư vấn về tiến trình của yêu cầu. Trong đáp ứng 2xx cho yêu cầu INVITE thân bản tin chứa miêu tả phiên. Trong đáp ứng 3xx, thân bản tin chứa miêu tả về đích chọn lựa hay dịch vụ. Với các đáp ứng 4xx và lớn hơn, thân bản tin gồm các thông tin bổ sung, các thông tin con người có thể đọc được để biết được về các nguyên nhân gây lỗi.

5.5.2 Kiểu và độ dài thân bản tin

Kiểu truyền thông của thân bản tin được chỉ ra bởi trường tiêu đề Content-Type. Nếu thân bản tin đã qua mã hóa thì nó được chỉ ra bởi trường tiêu đề Content-Encoding. Tập hợp các ký tự của thân bản tin được chỉ ra bởi giá trị trường tiêu đề Content-Type.

Độ dài của thân bản tin (Message Body Length) được tính theo byte và được đưa ra bởi trường tiêu đề Content-Length.

5.6 KHUÔN DẠNG BẢN TIN SIP

Khi SIP được truyền thông qua giao thức UDP với sự xác thực và một miêu tả phiên phức tạp thì độ dài của một yêu cầu hay một đáp ứng SIP sẽ lớn hơn MTU. Để giải quyết vấn đề này, khuôn dạng thoả thuận của SIP được định nghĩa bằng cách sử dụng các từ viết tắt cho các trường tiêu đề chung. Các định nghĩa viết tắt được miêu tả ở bảng 5.3.

Bảng 5.3:Các từ viết tắt cho tiêu đề chung

Tên trường ngắn	Tên trường dài
C	Content-Type
E	Content-Encoding
F	From
I	Call-ID
M	Contact
L	Content-Length
S	Subject
T	To
V	Via

Ở đây, Client có thể dùng cả tên trường ngắn và tên trường dài trong cùng một yêu cầu và Server cũng chấp nhận điều đó.

5.7 HOẠT ĐỘNG CỦA CÁC THÀNH PHẦN SIP

Như đã trình bày ở đầu chương này, khi xét trên quan điểm Client/Server thì hệ thống SIP gồm các thành phần chính như hình 5.1. Mục này tiến hành phân tích quá trình xử lý, hoạt động của các thành phần trong hệ thống SIP.

5.7.1 Hoạt động của SIP Client và SIP Server

SIP có thể sử dụng cả hai giao thức TCP và UDP làm giao thức truyền. Tuỳ theo loại dữ liệu và yêu cầu về độ tin cậy của phiên truyền thông đó mà TCP hay UDP được sử dụng.

Yêu cầu (Request)

Server loại bỏ những yêu cầu giống nhau và truyền lại các đáp ứng thích hợp. Sau khi nhận được một yêu cầu CANCEL từ thiết bị Upstream Client, một Proxy Server sẽ gửi một yêu cầu CANCEL tới tất cả các nơi mà nó chưa nhận được đáp ứng cuối cùng.

Khi một UA nhận được một yêu cầu, nó sẽ kiểm tra lại trường Call-ID dựa trên cuộc gọi đang diễn ra. Nếu như Call-ID được tìm thấy nó sẽ so sánh giá trị thẻ (tag) của trường "To" với thẻ của User. UA sẽ từ chối yêu cầu nếu như hai giá trị đó không giống nhau. Nếu giá trị của tiêu đề From (kể cả giá trị tag) giống với giá trị From của cuộc gọi hiện tại, Server sẽ so sánh giá trị của trường tiêu đề "CSeq". Nếu giá trị đó nhỏ hơn hoặc bằng giá trị CSeq hiện tại thì yêu cầu đó là một yêu cầu phát lại. Nếu giá trị đó lớn hơn thì đó là một yêu cầu mới. Nếu như tiêu đề "From" không giống với tiêu đề "From" của cuộc gọi hiện tại, có nghĩa một cuộc gọi mới được thiết lập.

Nếu giá trị Call-ID không tìm thấy, một cuộc gọi mới được thiết lập với các giá trị được ghi vào các trường tiêu đề To, From và Call-ID. Trong trường hợp này, trường tiêu đề To không chứa một thẻ (tag) địa chỉ nào. Server trả lại một đáp ứng chứa cùng một giá trị của trường To nhưng chỉ một thẻ địa chỉ duy nhất được thêm vào. Thẻ địa chỉ có thể được bỏ qua nếu yêu cầu chỉ chứa một trường tiêu đề Via.

Đáp ứng (Response)

Một Server có thể đưa ra một hay nhiều đáp ứng tạm thời trước khi nó gửi đáp ứng cuối cùng. Trong khoảng thời gian 200 ms, nếu một Stateful Proxy, UAS, Redirect Server hay Registrar Server không thể đưa ra một đáp ứng cuối cùng cho yêu cầu, ngay lập tức nó sẽ phát ra một đáp ứng tạm thời loại 1xx. Stateless Proxy không thể tự phát các đáp ứng tạm thời cho bản thân chúng.

Các đáp ứng được ánh xạ đến các yêu cầu bằng cách so sánh giá trị các trường tiêu đề To, From, Call-ID, CSeq và các thông số phụ của trường tiêu đề Via đầu tiên. Nếu như trường tiêu đề Via chỉ ra rằng Upstream Server sử dụng giao thức truyền TCP thì Proxy phải nhanh chóng mở một kết nối TCP tới địa chỉ đó. Như vậy, các Proxy phải được xử lý đặc biệt để nhận các yêu cầu đến từ các kết nối TCP thụ động mặc dù hầu hết các đáp ứng đều sẽ đến một kết nối TCP chủ động. Một kết nối TCP chủ động (Active TCP Connection) là một kết nối TCP được khởi tạo bởi Proxy còn một kết nối TCP thụ động là một kết nối được chấp nhận bởi Proxy nhưng được khởi tạo bởi thực thể khác.

Đáp ứng 100 không được gửi đi còn các đáp ứng 1xx khác có thể được gửi sau khi Server loại bỏ các đáp ứng mà mã trạng thái đã được gửi đi trước đó. Đáp ứng 2xx được

gửi đi theo tiêu đề Via. Khi Stateful Proxy nhận được một đáp ứng 2xx nó phải gửi đi một đáp ứng cuối cùng không phải là loại 2xx. Các đáp ứng với trạng thái 300 và lớn hơn được phát lại bởi các Stateful Proxy cho đến khi upstream Proxy tiếp theo gửi đi một yêu cầu ACK hay CANCEL.

Một Stateful Proxy phải duy trì trạng thái ít nhất là 32 giây kể từ khi nhận được đáp ứng Non-200 đầu tiên để điều khiển việc phát lại các đáp ứng. 32 giây là khoảng thời gian phát lại lớn nhất của một đáp ứng 200, được dùng mặc định trong trường hợp yêu cầu ACK bị mất trên đường tới UA bị gọi hoặc Stateful Proxy tiếp theo.

Địa chỉ nguồn, địa chỉ đích và các kết nối

Kết nối UDP đơn hướng: các đáp ứng được trả lại địa chỉ được chỉ ra trong trường tiêu đề Via (không có địa chỉ nguồn của đáp ứng). Với cuộc gọi về các đáp ứng không được phát ra bởi Next-hop Stateless Server mà được phát ra bởi một Proxy Server hoặc UAS. Do đó, Stateless Proxy có thể chỉ sử dụng trường tiêu đề Via để gửi các đáp ứng.

Kết nối UDP đa hướng: yêu cầu đa hướng có thể đặc trưng cho Host và không phụ thuộc vào trường Request-URI. Yêu cầu đa hướng phải đảm bảo rằng nó không ra khỏi phạm vi của hệ thống quản lý. Điều này được thực hiện bởi TTL hay phạm vi quản lý tùy theo sự thực hiện của nó trong mạng. Một Client nhận được một yêu cầu đa hướng không phải kiểm tra xem thành phần Host của trường Request-URI có giống với Host hay tên vùng của nó không. Nếu yêu cầu nhận được là yêu cầu đa hướng thì đáp ứng trở lại cũng phải là đáp ứng đa hướng. Đáp ứng cho các yêu cầu đa hướng được phát nhiều hướng với cùng thông số TTL như trong yêu cầu. TTL được lấy ra từ thông số ttl trong trường tiêu đề Via.

Kết nối TCP: một kết nối TCP đơn có thể phục vụ một hay nhiều giao dịch SIP. Một giao dịch SIP không chứa hay chứa nhiều đáp ứng tạm thời theo sau là một hay nhiều đáp ứng cuối cùng. Client có thể kết nối khi mà đáp ứng cuối cùng đầu tiên tới. Nếu Client đóng hoặc xác lập lại kết nối TCP ưu tiên để nhận được các đáp ứng cuối cùng đầu tiên, Server sẽ coi hoạt động này như một yêu cầu CANCEL. Hoạt động này phải được hạn chế vì có thể khiến Client làm việc sai chức năng dẫn đến một Proxy Server sẽ chiếm giữ một kết nối vô thời hạn. Server không được đóng kết nối TCP cho đến khi nó gửi đi một đáp ứng cuối cùng. Tuy nhiên, bình thường thì Client chịu trách nhiệm đóng kết nối. Nếu như một Server muốn trả lại một đáp ứng cho Client mà không muốn mở một kết nối dành cho Client đó thì nó sẽ mở một kết nối khác tới danh sách địa chỉ trong trường Via. Theo cách đó một Proxy hay UA phải được xử lý đặc biệt để nhận cả các yêu cầu và đáp ứng trong cùng một kết nối thụ động.

5.7.2 Hoạt động của Proxy Server, Location Server và Redirect Server

Proxy Server

Proxy Server có thể kết nối theo kiểu nhánh, một yêu cầu vào có thể cho sinh ra nhiều yêu cầu ra tới Client. Proxy Server có thể là một Stateful Proxy hay là một Stateless Proxxy. Khi là Stateful, Proxy Server thực hiện nhận một yêu cầucho việc tạo ra và phát đi các yêu cầu sinh ra từ yêu cầu đó. Nếu là Stateless, Proxy Server bỏ qua tất cả các thông tin mỗi khi yêu cầu được phát.

Stateful Proxy hoạt động như một UAS/UAC ảo. Nó thực hiện chức năng Server khi nhận các yêu cầu và thực hiện chức năng Client cho việc phát đi các yêu cầu, ngoại trừ việc nhận đáp ứng 2xx cho yêu cầu INVITE.

Stateless Proxy hoạt động không giống như UAS hay UAC ảo. Stateless Proxy chuyển các yêu cầu nó nhận được từ các thực thể Downstream (đường xuống) và chuyển các đáp ứng mà nó nhận được từ thực thể Upstream (đường lên).

1. Proxy Request

Để tránh lặp lại, một Server phải kiểm tra xem địa chỉ của nó có thực sự nằm trong trường tiêu đề Via của yêu cầu đến không. Các giá trị của trường To, From, Call-ID và Contact đều được sao chép chính xác từ yêu cầu ban đầu. Proxy có thể thay đổi trường Request-URI để thông báo cho Server nơi nó muốn gửi yêu cầu đến. Một Proxy Server bao giờ cũng chèn một trường tiêu đề Via chứa địa chỉ của chính nó vào các yêu cầu.

2. Proxy Response

Proxy chỉ xử lý một đáp ứng khi mà trường Via cao nhất phù hợp với một trong các địa chỉ của nó. Nếu không đáp ứng đó sẽ bị dừng lại có nghĩa nó không được xử lý.

3. Stateless Proxy

Một Stateless Proxy sẽ loại bỏ trường Via của chính nó và kiểm tra địa chỉ trường Via tiếp theo. Trong trường hợp giao thức truyền là UDP, đáp ứng được gửi đến địa chỉ được liệt kê trong thẻ "maddar" nếu chúng có mặt, nếu không nó gửi đến địa chỉ liệt kê trong trường "received" nếu chúng có mặt và cuối cùng được gửi đến địa chỉ trong trường "sent-by". Proxy vẫn giữ nguyên trạng thái "statefull" khi thiết lập yêu cầu nhận thông qua TCP. Stateless Proxy không được phát ra yêu cầu tạm thời.

4. Statefull Proxy

Khi một Statefull Proxy nhận được một yêu cầu, nó sẽ kiểm tra trường To, From, Call-ID và CSeq dựa vào bản ghi của yêu cầu hiện tại. Nếu không có mặt yêu cầu tương

ứng với một giao dịch mới và yêu cầu được ủy quyền. Một Statefull Proxy Server có thể phát đáp ứng tạm thời 1xx.

Location Server

Location Server thực hiện việc định vị thuê bao, nó cung cấp vị trí của các thành viên tham gia phiên truyền thông. Vị trí đó được Location Server cung cấp cho các thành phần khác của hệ thống SIP như Proxy Server và Redirect Server.

Redirect Server

Một Redirect Server không thể phát ra các yêu cầu SIP cho chính nó. Sau khi nhận được một yêu cầu khác yêu cầu CANCEL, nó thu thập các vị trí thay đổi nhờ thông tin từ Location Server và trả lại một đáp ứng cuối cùng loại 3xx hoặc là từ chối yêu cầu. Với các yêu cầu CANCEL có khuôn dạng hợp lệ, nó có thể trả lại một đáp ứng 2xx. Đáp ứng này sẽ kết thúc phiên giao dịch SIP. Redirect Server sẽ duy trì trạng thái giao dịch trong suốt phiên giao dịch SIP.

5.7.3 Hoạt động của UA (User Agent)

Phần này mô tả các nguyên tắc mà UAC và UAS dùng để tạo ra và xử lý các yêu cầu hay đáp ứng.

Chủ gọi phát yêu cầu INVITE

Khi UAC muốn khởi tạo một cuộc gọi, nó sẽ đưa ra một yêu cầu INVITE. Trường To trong yêu cầu chứa địa chỉ của bị gọi, trường Request-URI chứa cùng địa chỉ đó. Trường From chứa địa chỉ của chủ gọi. Nếu địa chỉ From có thể xuất hiện trong yêu cầu phát ra từ các UAC khác cho cùng một cuộc gọi thì phía chủ gọi phải đưa thêm thông số "tag" vào trong trường From. UAC có thể nhập thêm trường Contact chứa một địa chỉ mà nó muốn kết nối sau khi phía bị gọi giải phóng hội thoại với chủ gọi.

Bị gọi trả lời

Khi bị gọi nhận được yêu cầu INVITE, thì bị gọi có thể chấp nhận, gửi lại hay từ chối cuộc gọi. Trong tất cả các trường hợp đó nó đều phải phát ra đáp ứng. Đáp ứng này phải lấy các giá trị trong trường To, From, Call-ID, CSeq và Via từ yêu cầu. Thêm vào đó, đáp ứng UAS phải bổ sung tham số "tag" vào trường To của đáp ứng nếu yêu cầu chứa nhiều hơn một trường tiêu đề Via. UAS có thể thêm một trường tiêu đề Contact vào đáp ứng. Nó chứa một địa chỉ mà phía bị gọi muốn kết nối cho giao dịch tiếp theo bao gồm cả ACK cho INVITE hiện tại. UAS lưu giữ giá trị của trường To và From kể

cả các thẻ "tag". Chúng sẽ trở thành địa chỉ nội hạt hay địa chỉ xa của các cuộc gọi tương ứng.

Chủ gọi nhận đáp ứng ban đầu của yêu cầu

Đối với một yêu cầu INVITE có rất nhiều đáp ứng có thể đến cùng một UAC. Các đáp ứng được phân biệt nhau bởi tham số "tag" trong trường tiêu đề To đại diện cho từng cuộc gọi riêng. Chủ gọi có thể chấp nhận hay kết thúc với mỗi đáp ứng UAS. Để chấp nhận nó gửi một yêu cầu ACK còn để kết thúc nó gửi một yêu cầu BYE. Trường tiêu đề To và From trong yêu cầu ACK hay BYE phải giống trường tiêu đề To và From trong đáp ứng 200 OK (bao gồm cả giá trị tag). Trường Request-URI của ACK hay BYE có thể thiết lập bất cứ địa chỉ nào được tìm thấy trong trường tiêu đề Contact của đáp ứng 200 OK nếu nó tồn tại. Lần lượt UAC sẽ sao chép các địa chỉ từ trường tiêu đề To vào Request-URI. UAC cũng lưu ý đến giá trị của trường tiêu đề To và From trong mỗi đáp ứng. Với mỗi cuộc gọi trường tiêu đề To sẽ trở thành địa chỉ xa còn trường tiêu đề From sẽ là địa chỉ nội hạt.

Các yêu cầu đưa ra tiếp theo

Khi một cuộc gọi được thiết lập cả phía chủ gọi hoặc bị gọi đều có thể phát ra yều cầu INVITE hay BYE để thay đổi hay kết thúc cuộc gọi. Không chú ý đến bị gọi hay chủ gọi phát yêu cầu mới, trường tiêu đề trong yêu cầu được thiết lập như sau: Với mỗi cuộc gọi giá trị trường tiêu đề To sẽ được thiết lập thành địa chỉ xa còn giá trị trường tiêu đề From sẽ thành địa chỉ nội bộ. Trường tiêu đề Contact có thể khác với trường tiêu đề Contact được gửi đi trong yêu cầu hay đáp ứng trước đó. Trường Request-URI thiết lập với giá trị của trường tiêu đề Contact nhận được trong một yêu cầu hay đáp ứng trước đó từ một thành viên xa hoặc tới giá trị của một địa chỉ xa.

Khi nhận được một yêu cầu, các kiểm tra sau được thực hiện:

- Nếu như Call-ID là mới, yêu cầu đó là cho cuộc gọi mới. Không chú ý đến giá trị của trường tiêu đề To và From.

- Nếu Call-ID đã tồn tại, yêu cầu đó là dành cho cuộc gọi hiện tại. Nếu các giá trị To, From, Call-ID và CSeq thực sự phù hợp với các yêu cầu nhận được trước đó (kể cả giá trị tag) thì yêu cầu đó là một yêu cầu phát lại.

- Nếu không có sự phù hợp với các bước trước đó, trường tiêu đề To và From được so sánh lại với cuộc gọi nội hạt hiện tại và cuộc gọi ở xa. Nếu phù hợp và giá trị trong CSeq trong yêu cầu không lớn hơn giá trị CSeq nhận được trong cuộc gọi trước đây, yêu cầu đó là một giao dịch mới cho cuộc gọi hiện tại.

Chương 6

GIAO THỨC ĐIỀU KHIỂN CỔNG PHƯƠNG TIỆN - MEGACO/H.248

6.1 GIỚI THIỆU CHUNG VỀ GIAO THỨC ĐIỀU KHIỂN CỔNG PHƯƠNG TIỆN

Đầu năm 1998 nhiều nhà nghiên cứu cũng như các nhà điều hành mạng đã đưa ra một số giao thức điều khiển cổng phương tiện khác nhau (MGCP của IETF, MDCP của ITU-T...). Mặc dù nhu cầu về một giao thức điều khiển cổng phương tiện là rất rõ ràng nhưng không có giao thức nào nhận được sự ủng hộ hoàn toàn. Tới cuối năm 1998, IETF thành lập nhóm nghiên cứu về giao thức điều khiển cổng phương tiện MEGACO (MEgaco GAteway Control working group) với nhiệm vụ xây dựng một tiêu chuẩn mở trên cơ sở IP dựa trên nguyên lý chủ/tớ để điều khiển các cổng phương tiện, Nortel Network là tổ chức đứng đầu nhóm này.

Nhóm nghiên cứu đã lựa chọn các ưu điểm của hai giao thức tiền thân MGCP và MDCP, đồng thời bổ sung các tính năng tăng cường, tới tháng 3 năm 1999, Nortel Network chính thức công bố về một giao thức điều khiển cổng phương tiện mới MEGACO.

Song song với các nỗ lực của IETF trong việc xây dựng giao thức MEGACO, vào tháng 5 năm 1999 nhóm nghiên cứu 16 (SG16 - study group 16) của ITU-T cũng khởi xướng một dự án xây dựng giao thức điều khiển cổng phương tiện GCP, sau này là giao thức H.248. Cũng vào mùa hè năm đó, hiệp định về việc xây dựng một chuẩn quốc tế MEGACO/H.248 cũng được ký kết giữa hai tổ chức này. Tới tháng 6 năm 2000, MEGACO/H.248 chính thức được phê chuẩn bởi hai tổ chức IETF và ITU-T.

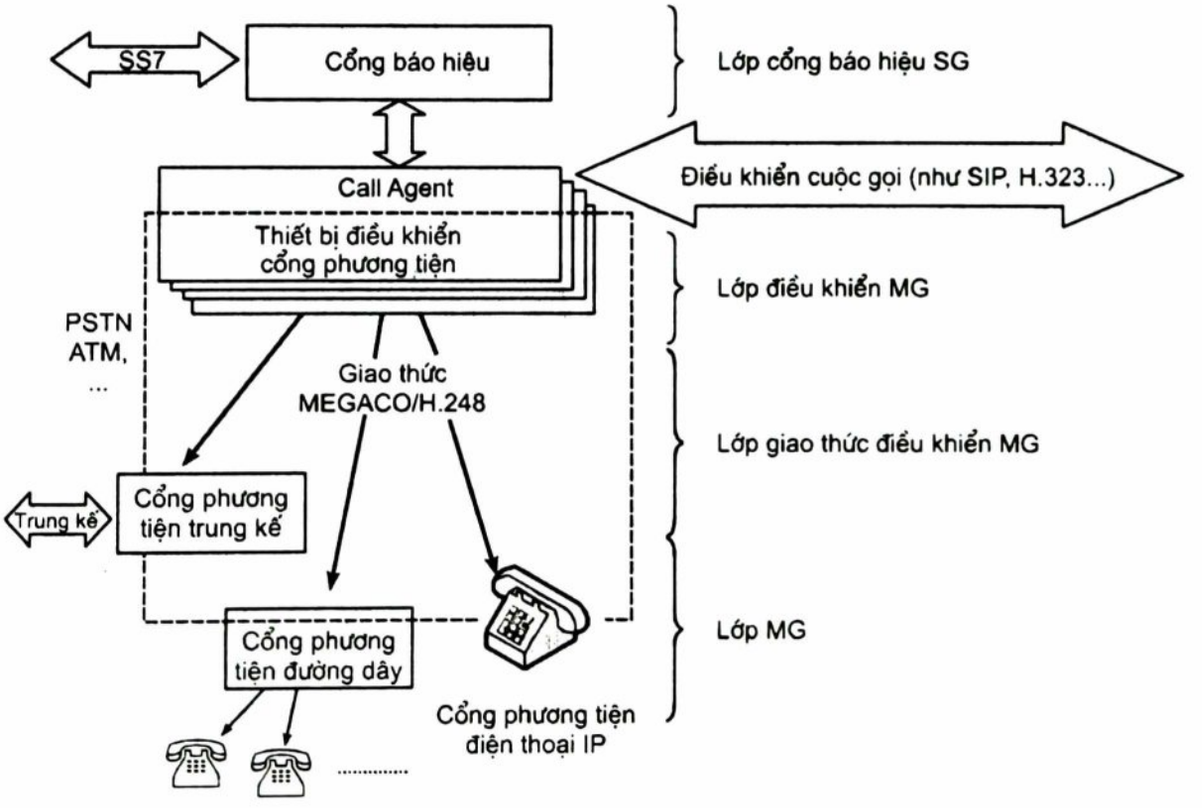
Trong quá trình xây dựng MEGACO/H.248, những người ủng hộ MGCP vẫn yêu cầu IETF đưa ra một RFC về giao thức MGCP. Vào tháng 10 năm 1999, văn bản này được ban hành nhưng đây không phải là tiêu chuẩn chính thức của IETF, RFC này cũng không được SG16 của ITU-T thừa nhận mặc dù hiện nay ISC ủng hộ MGCP.

Vào thời điểm hiện nay MEGACO là giao thức chuẩn quốc tế duy nhất dành cho điều khiển mọi loại cổng phương tiện theo cơ chế chủ/tớ. MEGACO sẽ tiếp tục được cải tiến và phát triển hơn nữa nhờ nỗ lực không ngừng của IETF, ITU-SG16 và các tổ chức tiêu chuẩn hóa khác.

6.2 GIAO THỨC MEGACO/H.248

6.2.1 Tổng quan về giao thức MEGACO/H.248

MEGACO là giao thức điều khiển cổng phương tiện nói chung, bao gồm cổng nội hạt, trung kế trong mạng PSTN, giao diện ATM, giao diện thoại và đường dây tương tự, điện thoại IP, các loại máy chủ... Với tính năng hỗ trợ rộng rãi các ứng dụng một cách mềm dẻo, đơn giản và hiệu quả ở mức chi phí hợp lý, MEGACO sẽ là chuẩn được sử dụng trong mạng thế hệ sau NGN. MEGACO không bị ràng buộc với bất kỳ một giao thức điều khiển cuộc gọi ngang cấp nào (ví dụ như SIP hay H.323) và hoàn toàn tùy thuộc vào thiết kế của người quản trị mạng. Kiến trúc của MEGACO dựa trên 3 lớp: lớp MGC, lớp MG và lớp MEGACO (Hình 6.1).



Hình 6.1 Kiến trúc điều khiển của MEGACO/H.248

• Lớp MGC chứa tất cả các phần mềm điều khiển, xử lý cuộc gọi. Lớp này thực hiện các đặc điểm ở mức cuộc gọi như phát triển cuộc gọi, chuyển cuộc gọi, hội thoại hay giữ cuộc gọi. Lớp MGC cũng thực hiện giao tiếp với các MGC cũng như các thực thể ngang cấp hay cấp dưới khác, MGC quản lý mọi thuộc tính trong quá trình giao tiếp.

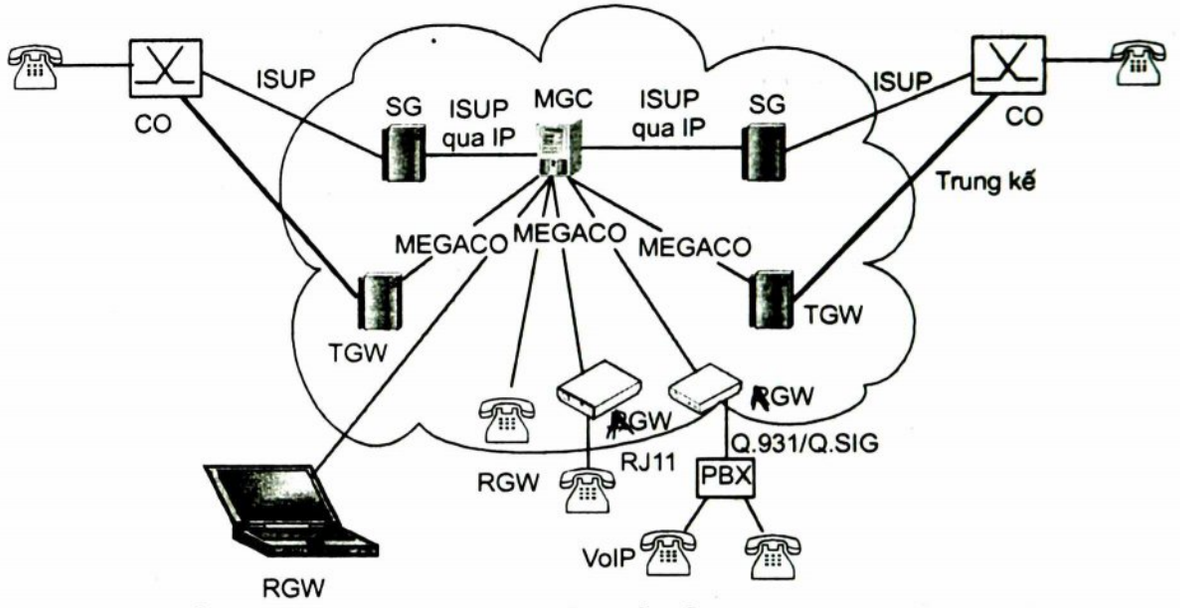
• Lớp MG thực hiện các kết nối lưu lượng đi và tới các mạng khác, tương tác với các luồng lưu lượng này qua ứng dụng báo hiệu và sự kiện. Lớp MG cũng điều khiển

các thuộc tính thiết bị của cổng phương tiện (ví dụ như giao diện với người dùng). Lớp này không hề biết gì về việc điều khiển các thuộc tính cuộc gọi và hoạt động theo sự điều khiển của lớp MGC.

- Lớp MEGACO/H.248 quy định cách thức mà lớp MGC điều khiển lớp MG.

6.2.2 Chức năng của giao thức MEGACO/H.248

Giao thức MEGACO/H.248 định nghĩa giao diện điều khiển của MGC đối với MG. MEGACO cung cấp các chức năng sau:

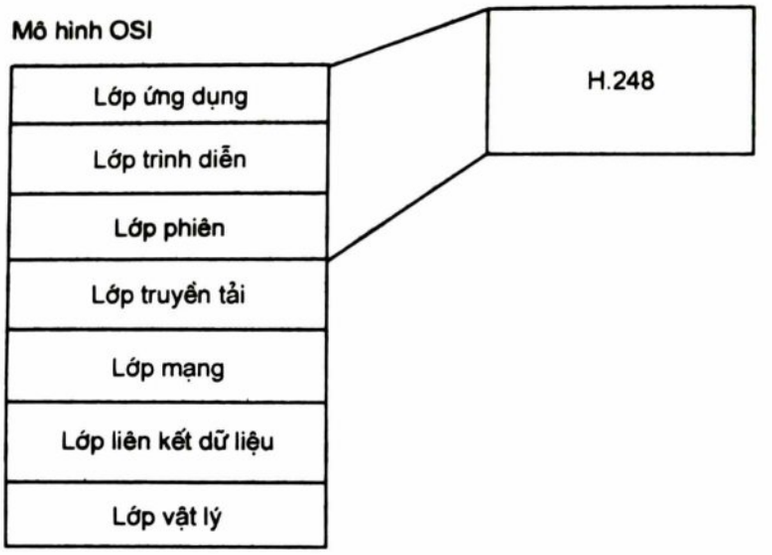


Hình 6.2: Vị trí và chức năng của giao thức MEGACO/H.248

- Điều khiển các loại MG khác nhau (TGW, RGW, AGW, MS...)
- Hỗ trợ đàm phán quyết định các thuộc tính cuộc gọi
- Có khả năng xử lý cuộc gọi đa người dùng
- Hỗ trợ QoS và đo lường lưu lượng (các thông tin thống kê sau mỗi kết nối)
- Thông báo lỗi giao thức, mạng, hay các thuộc tính cuộc gọi

6.2.3 Vị trí của giao thức MEGACO/H.248 trong mô hình OSI

Như chỉ ra trong hình 6.3 giao thức MEGACO thực hiện chức năng của mình ở 3 lớp trên cùng trong mô hình OSI: lớp ứng dụng, lớp trình diễn và lớp phiên.



Hình 6.3: Giao thức MEGACO trong mô hình OSI

6.2.4 Các khái niệm trong giao thức MEGACO/H.248

Các loại cổng (GateWay): GW có địa chỉ IP và số cổng UDP mặc định của MGC quản lý nó

- TGW (Trunking GateWay) là cổng nối giữa các tổng đài trung kế mạng PSTN và mạng IP.
- AGW (Access GateWay) là cổng nối giữa mạng ISDN, tổng đài PBX và mạng IP.
- RGW (Residential GateWay) là cổng nối giữa một nhóm các máy điện thoại (nối trực tiếp vào cổng này) và mạng IP.
- NAS (Network Access Server) cung cấp khả năng truy nhập Internet.
- IVR (Interactive Voice Respone) được MGC điều khiển để thu các số và gửi FAX...

Mỗi loại GW có khả năng hỗ trợ các gói tin khác nhau.

Thiết bị đầu cuối (Termination): MGC coi GW là đại diện cho một nhóm các termination, trong đó mỗi termination chịu trách nhiệm xử lý cho một loại lưu lượng. Mỗi termination được GW khởi gán một ID tại thời điểm nó được tạo ra. Có hai loại termination là:

- Termination cố định (các giao diện vật lý, kênh TDM). Đối với loại termination này việc sử dụng các lệnh "Add" hay "Subtract" chỉ đơn giản là việc lấy chúng ra hay thêm vào Null context.
- Termination tạm thời hay termination lô-gíc (cổng RTP nối cổng nối với mạng, chỉ tồn tại khi sử dụng). Loại termination này được tạo ra bởi lệnh "Add" và bị xóa đi bởi lệnh "Subtract".

GW được MGC coi như termination gốc, điều này có ý nghĩa khi MGC muốn làm việc với chính GW (ví dụ khi tuyên bố GW là "in" hay "out" một dịch vụ nào đó). Giao thức MEGACO có khả năng làm việc với một số lượng lớn các termination là do termination có các thuộc tính lựa chọn (event, signal, statistic). Các thuộc tính này được đưa vào các gói và MGC có thể chỉ định termination là nó chỉ tiếp nhận những gói nào. Tại mỗi thời điểm, termination cũng được điều khiển ở một chế độ xác định (chế độ chỉ nhận, chỉ gửi hoặc vừa nhận vừa gửi). Termination là nơi đi và đến của các luồng lưu lượng hay điều khiển.

Context: Là khái niệm mang tính đột phá của MEGACO so với các giao thức cùng loại trước nó. Khái niệm này cho phép tạo ra các phiên liên lạc đa điểm. Mỗi đầu cuối có thể tham gia nhiều context khác nhau với các loại lưu lượng khác nhau (ví dụ khi đầu cuối tham gia một phiên truyền thông đa phương tiện). Đây là một phiên kết nối bao gồm nhiều termination. Mỗi context được GW tạo ra ban đầu chỉ có một termination, các termination kết nối với termination đầu tiên sẽ lần lượt được thêm vào context này. Context sẽ bị xoá bỏ khi termination cuối cùng được giải phóng. Số lượng termination tối đa trong một context phụ thuộc vào khả năng của GW (những GW chỉ hỗ trợ liên lạc điểm - điểm sẽ chỉ có tối đa 2 termination trong một context). Context được MG gán một số nhận dạng ID bao gồm 32 bit (ID là duy nhất đối với một MG). Tất cả các termination không tham gia vào một kết nối nào sẽ nằm trong Null Context. Hiện nay Context có 3 thuộc tính: topology descriptor (mô tả topo phiên kết nối), priority flag (cờ ưu tiên, hướng dẫn GW phân bổ tài nguyên trong trường hợp khan hiếm), emergency flag (cờ khẩn, chỉ ra context nào sẽ được lưu lại và khôi phục trong trường hợp có sự cố). Các thuộc tính khác của context có thể được bổ sung trong tương lai.

Transaction: Là một chuyển giao, mỗi transaction gồm transaction hỏi (Request) và transaction trả lời (Reply). Transaction Pending được sử dụng để thông báo rằng transaction vẫn đang được xử lý (sử dụng khi transaction nào đó bị timeout). Mỗi transaction có một số nhận dạng ID. Số nhận dạng ID có giá trị từ 1 tới 99999. Thông tin trao đổi giữa MGC và các GW dưới dạng các chuyển giao (chứa các lệnh và các bản tin thông báo sự kiện, trả lời...), các lệnh trong mỗi chuyển giao sẽ được xử lý theo thứ tự đã chỉ ra.

Event: Là các sự kiện của đầu cuối (như onhook (nhấc máy), offhook (đặt máy)...). Các sự kiện này được MG phát hiện và báo cáo tới MGC. MGC sẽ xem xét chỉ các sự kiện mà nó quan tâm ở bất kỳ thời điểm nào (được chỉ ra bởi event descriptor).

Signal: Là báo hiệu tạo ra các âm báo hay hiển thị hình ảnh ở đầu cuối. Thời gian tồn tại của báo hiệu có thể do MGC quy định hoặc có thể tồn tại cho tới khi bị loại bỏ. Nó sẽ bị loại bỏ bất cứ khi nào một sự kiện được phát hiện ở đầu cuối trừ khi có sự can thiệp của MGC.

Termination descriptor (miêu tả các kết cuối): Các thuộc tính của termination được nhóm lại thành các descriptor bao gồm:

- Termination state descriptor: Các thuộc tính của termination (test, có quyền hay bị cấm sử dụng một dịch vụ...)
- Media descriptor: Bảng mô tả các luồng lưu lượng (thoại, đặt máy,...)
- Event descriptor: Mô tả các sự kiện được phát hiện bởi GW và hành động đáp lại sự kiện đó
- Signal descriptor: Mô tả các báo hiệu có thể xảy ra ở termination
- Stream descriptor: Bảng các descriptor của remote, local, local control đối với một luồng lưu lượng xác định.
- Local control descriptor: Chứa các thuộc tính cần quan tâm giữa GW và MGC (chế độ của termination, các xử lý của GW khi nhận được local hay remote descriptor).
- Local descriptor: Chứa các thuộc tính chỉ ra luồng lưu lượng mà GW nhận từ thực thể đầu xa.
- Remote descriptor: Chứa các thuộc tính chỉ ra luồng lưu lượng mà GW gửi tới thực thể đầu xa.
- Digitmap descriptor: Định nghĩa một dãy tập hợp các sự kiện (về quay số) có thể có (theo kế hoạch đánh số). Dãy này được gửi tới GW để thu các chữ số quay bởi khách hàng và gửi tới MGC dưới dạng các gói hoàn chỉnh (toàn bộ số bị gọi). Ví dụ về một "digitmap":

(0| 00|[1-7]xxx|8xxxxxxx|Fxxxxxxx|Exx|91xxxxxxxxxx|9011x.)

Trong đó:

0	Local operator (Nhà khai thác nội hạt)
00	Long-distance operator (Nhà khai thác đường dài)
xxxx	Local extension number (starts with 1-7)
8xxxxxxx	Local number (số nội hạt)
#xxxxxxx	Off-site extension

*xx Star services

91xxxxxxxxxx Long-distance number (Số đường dài)

9011 + 15 số đi quốc tế

- Observed event descriptor: Mô tả các sự kiện quan sát được (có trong lệnh "***Notify***" hay "***Audit Value***")

- Package descriptor: Trong lệnh "***Audit Value***", dùng để trả lại bảng các gói tin gửi đi từ termination.

- Service change descriptor: Lý do đưa ra lệnh "***Service change***".

- Statistic descriptor: Trong lệnh "***Audit***" hay "***Subtract***", để báo cáo các thống kê ở termination.

- Topology descriptor: Chỉ ra topo (cấu hình) của context.

- Audit descriptor: Trong lệnh "***Audit***", xác nhận thông tin yêu cầu

- Error descriptor: Chứa mã lỗi và mã lỗi văn bản (tuỳ chọn) trong lệnh "***Notify***" (hỏi và trả lời).

- Event Buffer descriptor: Mô tả sự kiện vừa được GW phát hiện khi kích hoạt event buffering.

6.2.5 Truyền dẫn các bản tin của giao thức MEGACO/H.248

Cơ chế truyền dẫn sử dụng cần phải đảm bảo tính tin cậy, các lệnh đang gửi đi và chờ xác nhận từ phía nhận sẽ được lưu giữ một cách độc lập.

Các bản tin MEGACO có thể được truyền dẫn qua lớp UDP/IP hoặc TCP/IP. Các GW và các MGC sẽ được gán các địa chỉ IP, các luồng lưu lượng đi và đến sẽ qua các cổng UDP hay TCP được chỉ ra. Ví dụ như cổng dành cho lệnh "***Service Change request***" là 2944 khi sử dụng mã hoá văn bản và 2945 khi sử dụng mã hoá nhị phân (đối với cả UDP và TCP), các cổng này cũng được sử dụng khi không có cổng nào được chỉ ra. Các transaction Reply sẽ được gửi tới cùng với cổng mà transaction request được gửi đi.

Trong quá trình truyền dẫn các bản tin MEGACO cũng đặt ra một vấn đề về độ trễ của các bản tin, dẫn tới độ trễ của các lệnh và tính "hợp thời" của các xử lý tại GW hay MGC. Để giải quyết vấn đề này RFC 3525 cũng đưa ra một số quy tắc trong việc xử lý các bản tin MEGACO tại MGC.

6.2.6 Các lệnh của giao thức MEGACO/H.248

Giao thức MEGACO sử dụng 8 lệnh trong giao diện điều khiển giữa MGC và GW. Bao gồm:

- Add: Được sử dụng để thêm một termination vào context, cũng có thể để tạo một context (nếu đó là termination đầu tiên trong context này).

- Modify: Sử dụng để thay đổi thuộc tính, sự kiện hay các báo hiệu ở một termination.

- Subtract: Sử dụng để xóa một termination khỏi context, cũng có thể là xóa luôn cả context (nếu đó là termination cuối cùng trong context này).

- Move: Chuyển một termination từ một context này sang một context khác.

- Audit Value: Trả lại trạng thái hiện tại của termination (báo hiệu, sự kiện, thuộc tính, số liệu thống kê).

- Audit Capability: Trả lại tất cả các giá trị có thể có của termination (báo hiệu, sự kiện, thuộc tính, số liệu thống kê).

Các lệnh trên được sử dụng bởi MGC.

- Notify: GW sử dụng để báo cáo các sự kiện mà nó phát hiện được tới MGC.

- Service Change: Lệnh này được sử dụng bởi:

  + GW, để thông báo với MGC rằng một nhóm termination có ý định rời khỏi hay tham gia một dịch vụ nào đó.

  + GW, để đăng ký với MGC nó khởi động.

  + MGC, để tuyên bố một chuyển giao tới GW

  + MGC, để lệnh cho MG nào đó đưa một nhóm termination hay một termination tham gia hoặc ra khỏi một dịch vụ.

6.2.7 Mã hóa lệnh của giao thức MEGACO/H.248

Các bản tin MEGACO có thể được mã hoá bằng hai cách: mã hoá nhị phân (binary encoding) và mã hóa văn bản (text encoding).

Trong phương pháp mã hoá nhị phân, tiêu chuẩn ISO/ITU ASN.1 được sử dụng. ASN.1 là ngôn ngữ định nghĩa cách gửi dữ liệu giữa các hệ thống không giống nhau, nó định nghĩa ở các hệ thống cùng một cú pháp dữ liệu (trong các giao thức tầng ứng dụng). ASN.1 được viết bằng các ngôn ngữ khác nhau trong từng hệ thống, phù hợp với

từng hệ thống. Khi một hệ thống muốn gửi dữ liệu, hệ thống đó sẽ mã hoá dữ liệu cần gửi theo ASN.1, sau đó gửi đi, hệ thống nhận sẽ tiến hành giải mã theo chuẩn định sẵn ASN.1. Các luật mã hoá theo chuẩn ASN.1 bao gồm: BER (Basic Encoding Rule), CER (Canonial Encoding Rule), PER (Package Encoding Rule), DER (Distinguished Encoding Rule). Việc sử dụng luật mã hoá nào là tuỳ người thiết kế.

Trong phương pháp mã hoá văn bản, chuẩn ABNF được sử dụng (RFC2234). Có thể sử dụng 2 định dạng: rút gọn (compact text) và đầy đủ (pretty text). Chúng được mô tả như sau:

Rút gọn

```
!/1 [124.124.124.222]
T = 9998 {C = -{SC = ROOT {SV {MT = RS, AD = 55555, PF = ResGW/ 1, RE = "901 MG Cold Boot"}}}}
```

Đầy đủ

```
MEGACO/ 1   [124.124.124.222]
   Transaction = 9998  {
        Context = - {
          ServiceChange = ROOT {
             Services     {
                Method = Restart,
                ServiceChangeAddress = 55555,
                Profile  =  ResGW/ 1,
                Reason  =  "901 MG Cold Boot"
              }
          }
      }
  }
```

Cả hai định dạng đều có ưu và nhược điểm của mình, định dạng rút gọn cho bản tin có kích thước nhỏ hơn, thời gian mã hoá ngắn hơn tuy nhiên độ tin cậy không cao bằng định dạng đầy đủ.

6.2.8 Cú pháp lệnh của giao thức MEGACO/H.248

Để tiện phân tích ta có thể lấy một lệnh để minh họa, đây là lệnh từ MGC tới GW.

MGC to RGW2:

```
MEGACO/1 [216.33.33.61]: 27000
Transaction = 1240 {
 Context = 2 {
  Modify = TermB {
   Signals {cg/bt}
   Media {
      LocalControl {
```

```
            Mode = recvonly}
            }
      },
      Modify = EphB {
        Media {
          LocalControl {
          Mode = recvonly}
          }
        }
      }
    }
  }
```

Lệnh trong ví dụ trên bao gồm các phần sau:

- Địa chỉ IP của nơi gửi (MGC hay GW) 216.33.33.61 (IPv4)
- Số cổng nơi gửi 2700
- Số nhận dạng transaction 1240

+ Context ID

+ Context ID...

Mỗi Context sẽ có nhiều lệnh:

o Notify

o Audit

o Modify

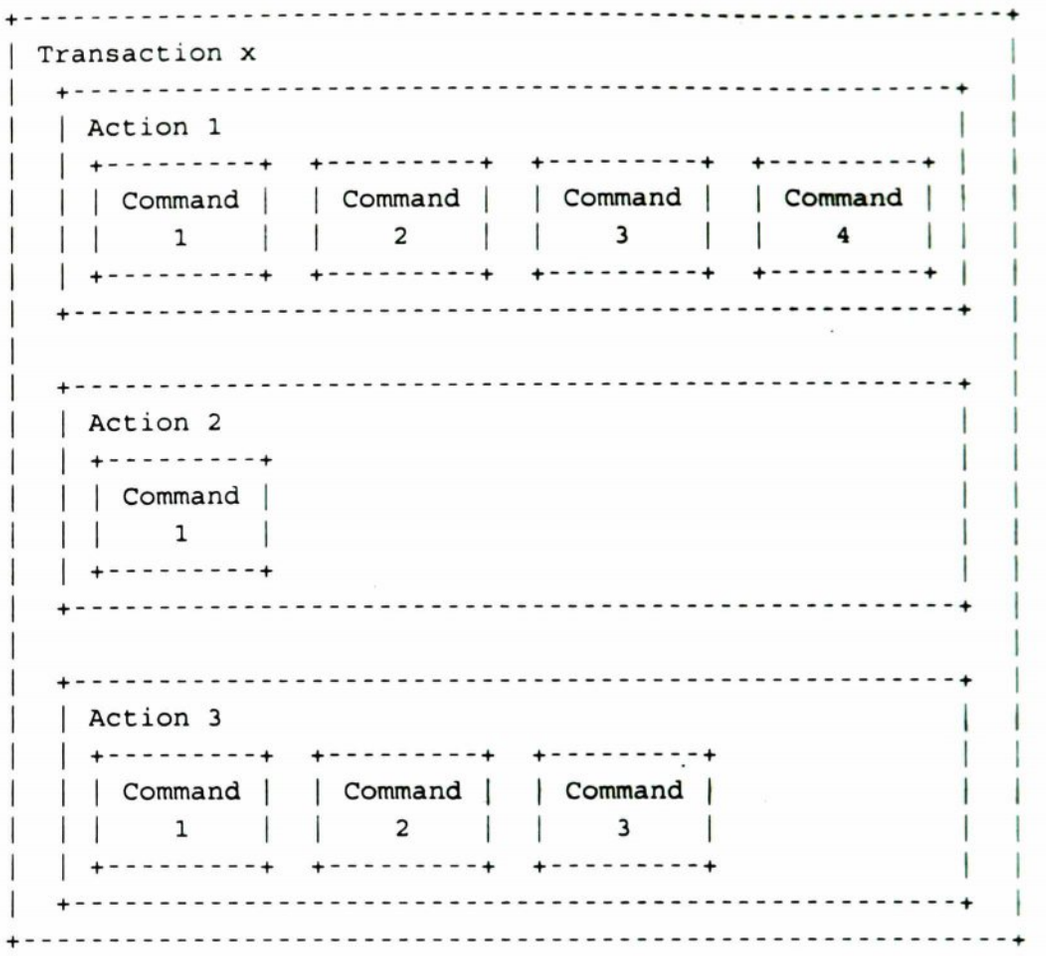
...

Trong mỗi lệnh lại gồm:

• Termination ID —TermB

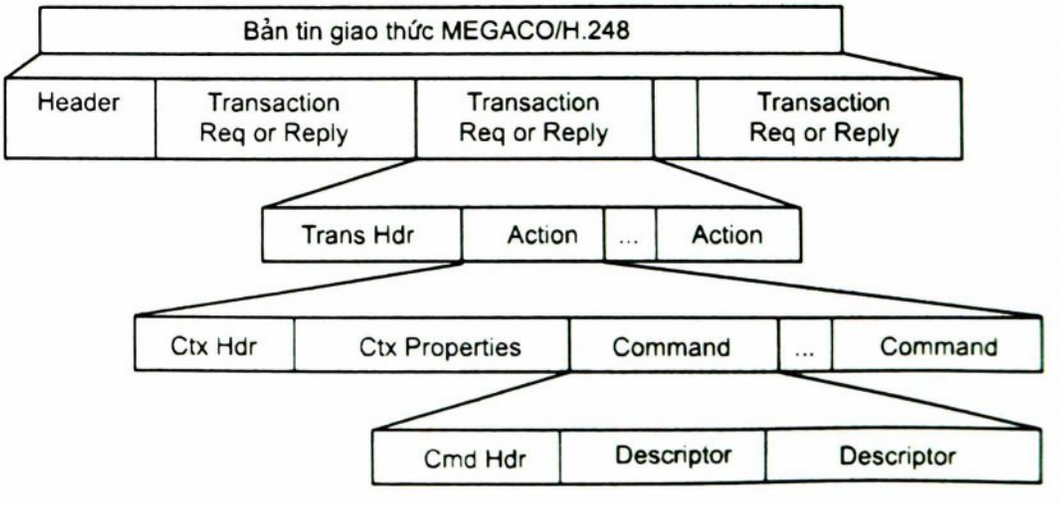
• Local Termination State

• Các descriptor



Ta có thể thấy: mỗi transaction gồm nhiều action (mỗi action cho 1 context), mỗi action gồm nhiều lệnh, mỗi lệnh sẽ có các descriptor.

6.2.9 Cấu trúc bản tin MEGACO/H.248



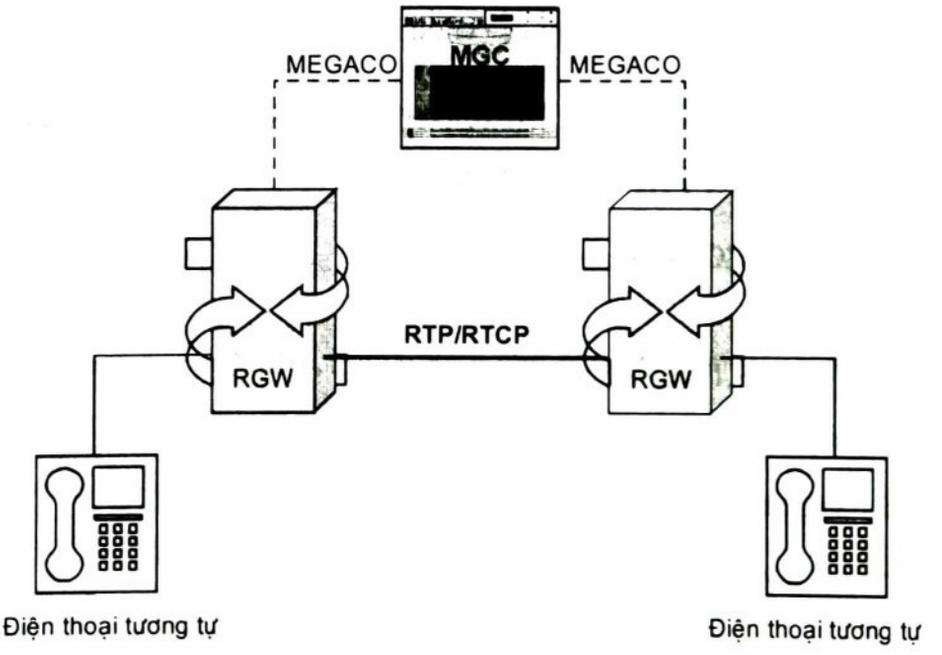
Hình 6.4: Cấu trúc bản tin MEGACO

Mỗi bản tin MEGACO bao gồm nhiều transaction và một header. Trong đó header chứa các thông tin nhận thực, bảo mật. Mỗi transaction gồm nhiều action (cho mỗi context) và một transaction header (chứa transaction ID). Mỗi context lại gồm một context header, một context properties và các lệnh. Context header chứa contextID, context properties chứa các thông tin về cấu hình context, thuộc tính, mức ưu tiên. Mỗi lệnh gồm một command header và các descriptor của lệnh đó.

6.3 HOẠT ĐỘNG CỦA GIAO THỨC MEGACO/H.248

Khi một đầu cuối nào đó nhấc máy và định thực hiện cuộc gọi, sự kiện nhấc máy này sẽ được phát hiện bởi MG quản lý nó. MG sẽ thông báo sự kiện này tới MGC trực thuộc, MGC sẽ chỉ định MG này bằng một lệnh để gửi âm báo mời quay số tới đầu cuối đó, đồng thời bản đồ số (digitmap) cũng được MG này cập nhật từ MGC, để phục vụ cho việc thu các chữ số và gửi toàn bộ số được quay về MGC.

Giả sử đầu cuối bị gọi thuộc một MG khác nhưng cùng được quản lý bởi MGC trên. Quá trình thiết lập liên kết được tiến hành theo 3 bước cơ bản sau:



Hình 6.5: Mô tả cuộc gọi MEGACO

- MGC yêu cầu MG thứ nhất thiết lập một kết nối tại điểm kết cuối thứ nhất. MG này sẽ phân bổ tài nguyên cho kết nối yêu cầu và đáp ứng lại bằng bản tin trả lời. Bản tin trả lời sẽ chứa các thông tin cần thiết để MG thứ hai có thể gửi các

bản tin một cách tin cậy tới liên kết vừa thiết lập. Các thông tin này có thể là: địa chỉ IP, tên cổng UDP, TCP hay các thông tin đóng gói bản tin.

- Tương tự, MGC cũng yêu cầu MG thứ hai thiết lập một liên kết ở điểm kết cuối thứ hai. MG này phân bổ tài nguyên cho kết nối này trên cơ sở các thông tin trong bản tin đáp ứng của MG thứ nhất. Tới lượt, MG thứ hai cũng đáp ứng lại bằng bản tin chứa các thông tin cần thiết nhằm đảm bảo MG thứ nhất có thể gửi các bản tin một cách tin cậy tới liên kết vừa thiết lập bởi MG thứ hai.
- Các thông tin trong bản tin đáp ứng của MG thứ hai sẽ được gửi tới MG thứ nhất. Khi này liên kết đã được thiết lập, quá trình truyền thông có thể diễn ra theo hai chiều. Lưu lượng được truyền tải nhờ các giao thức RTP hay RTCP.

Trong trường hợp hai MG được quản lý bởi 2 MGC khác nhau, các MGC này sẽ trao đổi các thông tin báo hiệu thông qua một giao thức báo hiệu từ MGC tới MGC (có thể là SIP hay H.323) để đảm bảo việc đồng bộ trong việc thiết lập kết nối tới hai điểm kết cuối.

Khi liên kết đã được thiết lập, các tham số của nó được giám sát bởi MGC và có thể được thay đổi dưới các lệnh của MGC (ví dụ như thêm một kết cuối vào liên kết).

6.4 ƯU ĐIỂM CỦA GIAO THỨC MEGACO/H.248 SO VỚI CÁC GIAO THỨC ĐIỀU KHIỂN CỔNG PHƯƠNG TIỆN KHÁC

So với giao thức MGCP, giao thức MEGACO version 1 (phiên bản 1) có các ưu điểm nổi trội sau:

- Hỗ trợ dịch vụ đa phương tiện, các dịch vụ hội nghị đa điểm tăng cường.
- Cải tiến cú pháp lệnh để việc xử lý bản tin hiệu quả hơn.
- Có khả năng lựa chọn giao thức TCP hay UDP.
- Chấp nhận cả việc mã hóa văn bản hay nhị phân.
- Các gói tin của MEGACO chi tiết hơn MGCP, hơn thế nữa các gói tin mới có thể được định nghĩa dựa trên các gói tin cơ sở này.
- Đưa ra khái niệm context, khái niệm này hỗ trợ kết nối đa dịch vụ, đa điểm.

Dưới đây sẽ mô tả hoạt động của MGC và MG theo giao thức MEGACO trong từng trường hợp cụ thể. Dưới đây sẽ khảo sát hoạt động của MEGACO hỗ trợ các dịch vụ thông thường và các dịch vụ bổ sung.

6.5 KẾT NỐI GIỮA HAI CỔNG NỘI HẠT (RGW)

Giả sử có hai đầu cuối người sử dụng được kết nối với hai RGW. Trong đó, hai RGW này được quản lý bởi cùng một MGC như biểu diễn hình 6.5. Quá trình MGC điều khiển các RGW xử lý cuộc gọi diễn ra như sau:

Bước 1: Ban đầu MGC gửi lệnh *"Modify"* tới tất cả các RGW để phát hiện sự kiện offhook (thuê bao nhấc máy). Bản tin của lệnh *"Modify"* như sau:

```
MGC to RGW1:
  MEGACO/1 [216.33.33.61]: 27000
  Transaction = 1234 {
    Context = - {
      Modify = TermA {
        Media {
            LocalControl {
              Mode = Receiveonly}
            },
        Events = 1111 {al/of}
      }
    }
  }
```

Bước 2: Các RGW lần lượt trả lời lệnh trên của MGC bằng các bản tin Reply

```
MG1 to MGC:
  MEGACO/1 [209.110.59.34]: 25000
  Reply = 1234 {
    Context = - {Modify = TermA}
  }
```

Bước 3: Giả sử người dùng A thuộc RGW1 nhấc máy tạo tín hiệu nhấc máy, sự kiện này sẽ được RGW1 báo cáo tới MGC bằng lệnh "*Notify*".

```
MG1 to MGC:
  MEGACO/1 [209.110.59.34]: 25000
```

```
Transaction = 2000 {
  Context = - {
    Notify = TermA {ObservedEvents =1111 {
      20010202T10000000:al/of}}
  }
}
```

Bước 4: MGC gửi Reply của lệnh này cho RGW1.

```
MGC to MG1:
  MEGACO/1 [216.33.33.61]: 27000
  Reply = 2000 {
    Context = - {Notify = TermA}
  }
```

Bước 5: MGC sẽ gửi lệnh *"Modify"* tới RGW1, lệnh này gồm 3 descriptor: signal descriptor được sử dụng để gửi âm mời quay số tới người dùng A, digitmap descriptor chứa mô hình mẫu các số có thể quay theo kế hoạch đánh số (ví dụ như 29****), event descriptor liệt kê các gói DTMF, gói tin hoàn thành quay số và gói tin giám sát trạng thái đặt máy của đầu cuối.

```
MGC to MG1:
  MEGACO/1 [216.33.33.61]: 27000
  Transaction = 1235 {
    Context = - {
      Modify = TermA {
        Signals {cg/dt},
        DigitMap= Dmap1 {(29XX)}
        Events = 1112 {
          al/on, dd/ce {DigitMap=Dmap1}
        },
      }}}
```

Bước 6: RGW1 trả lời MGC bằng một bản tin Reply. RGW1 tiến hành xử lý các descriptor theo thứ tự signal, digitmap, event descriptor. Đầu tiên âm mời quay số sẽ

được gửi tới termination A, sau đó digitmap sẽ được cập nhật vào cơ sở dữ liệu của RGW1, digitmap được kích hoạt khi RGW1 thu được sự kiện hoàn thành quay số.

termination A sau khi nhận được âm mời quay số sẽ tiến hành quay số.

```
MG1 to MGC:
  MEGACO/1 [209.110.59.34]: 25000
  Reply = 1235 {
     Context = - {Modify = TermA}
  }
```

Bước 7: Khi các con số được RGW1 thu đầy đủ và hợp lệ, chúng sẽ được gửi tới MGC bằng lệnh "*Notify*".

```
MG1 to MGC:
MEGACO/1 [209.110.59.34]: 25000
  Transaction = 2001 {
    Context = - {
      Notify = TermA {ObservedEvents =1112 {
        20010202T10010000:dd/ce {ds="2992", Meth=FM}}}
    }
  }
```

Bước 8: MGC xác nhận lệnh trên bằng bản tin Reply gửi tới RGW1.

MGC sau khi nhận lệnh trên sẽ phân tích số bị gọi và biết đầu cuối termination B đó thuộc RGW2 (giả sử đầu cuối này rỗi và sẵn sàng nhận cuộc gọi). MGC tiếp tục điều khiển RGW1.

```
MGC to MG1:
  MEGACO/1 [216.33.33.61]: 27000
  Reply = 2001 {
     Context = - {Notify = TermA}
  }
```

Bước 9: MGC sẽ gửi cho RGW1 hai lệnh. Lệnh "*Add*" để tạo một context và thêm ngay termination A vào đó. MGC biết rằng termination B rỗi, nó sẽ gửi hồi âm

chuông cho termination A. Lệnh thứ 2 để tạo một đầu cuối lô-gíc A và thêm đầu cuối này vào context vừa tạo ra.

```
MGC to MG1:
  MEGACO/1 [216.33.33.61]: 27000
  Transaction = 1236 {
    Context = $ {
      Add = TermA {
          Signals {cg/rt}
                }
      Add = $ {
        Media {
              {
          LocalControl {
            Mode = Receiveonly,
          },
          Local {
v=0
c=IN IP4 $
m=audio $ RTP/AVP 4
          }
        }
      }
    }
  }
```

Bước 10: RGW1 sẽ gửi bản tin Reply cho MGC bao gồm contextID(1), địa chỉ IP và số cổng dành cho lưu lượng.

```
MG1 to MGC:
  MEGACO/1 [209.110.59.34]: 25000
```

```
Reply = 1236 {
  Context = 1 {
    Add = TermA,
    Add=EphA{
      Media {
          Local {
v=0
c=IN IP4 209.110.59.33
m=audio 30000 RTP/AVP 4
a=recvonly
          } ; RTP profile for G.723 is 4
        }
      }
    }
  }
}
```

Bước 11: MGC sẽ gửi tới RGW2 2 lệnh. Lệnh 1 để tạo một context và thêm termination B vào context này. Báo hiệu chuông cũng được gửi tới termination B nhờ lệnh signal descriptor. Lệnh thứ hai sẽ tạo một đầu cuối lô-gíc B và thêm đầu cuối này vào context vừa tạo ra. Các thông tin địa chỉ IP, số cổng của termination A cũng được gửi tới RGW2.

```
MGC to MG2:
  MEGACO/1 [216.33.33.61]:27000
  Transaction = 1237 {
    Context = $ {
      Add = TermB  { Media {
          LocalControl {Mode = Receiveonly} },
        Signals {al/ri}
        Events=1234{al/of},
```

```
        },
      Add  = $ {Media {
            LocalControl {
              Mode = Receiveonly,
            },
            Local {
v=0
c=IN IP4 $
m=audio $ RTP/AVP 4
            },
            Remote {
v=0
c=IN IP4 209.110.59.33
m=audio 30000 RTP/AVP 4
            } ; RTP profile for G.723 is 4
          }
        }
      }
    }
  }
```

Bước 12: RGW2 sau khi nhận lệnh sẽ thực hiện lệnh và gửi kết quả thực hiện tới MGC, bao gồm contextID(2), địa chỉ IP và số cổng dành cho lưu lượng. MGC đợi cho termination B phát tín hiệu nhấc máy (offhook).

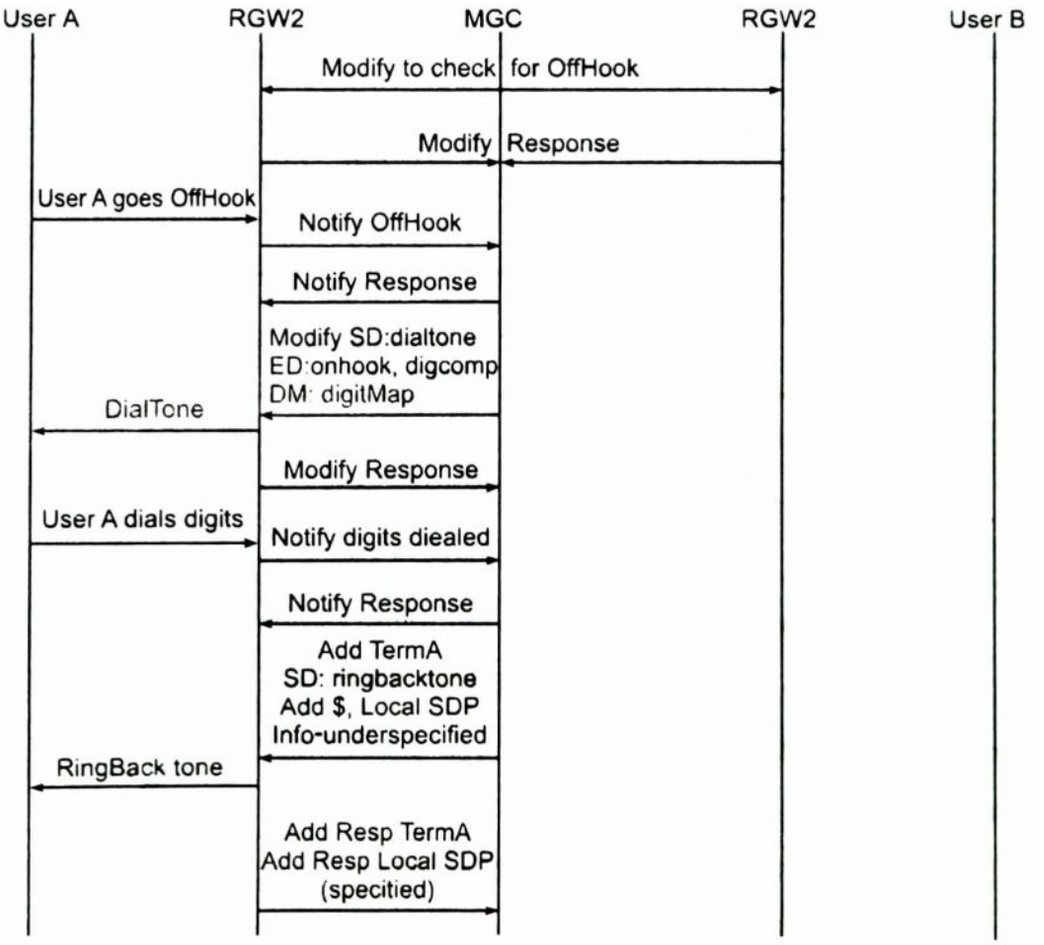
```
MG2 to MGC:
 'MEGACO/1 [207.176.47.89]: 26000
 Reply = 1237 {
   Context = 2 {
    Add = TermB,
    Add = EphB{
      Media {
           Local {
```

```
v=0
c=IN IP4 207.176.47.90
m=audio 40000 RTP/AVP 4
}
        } ; RTP profile for G723 is 4
    }
  }
}
```

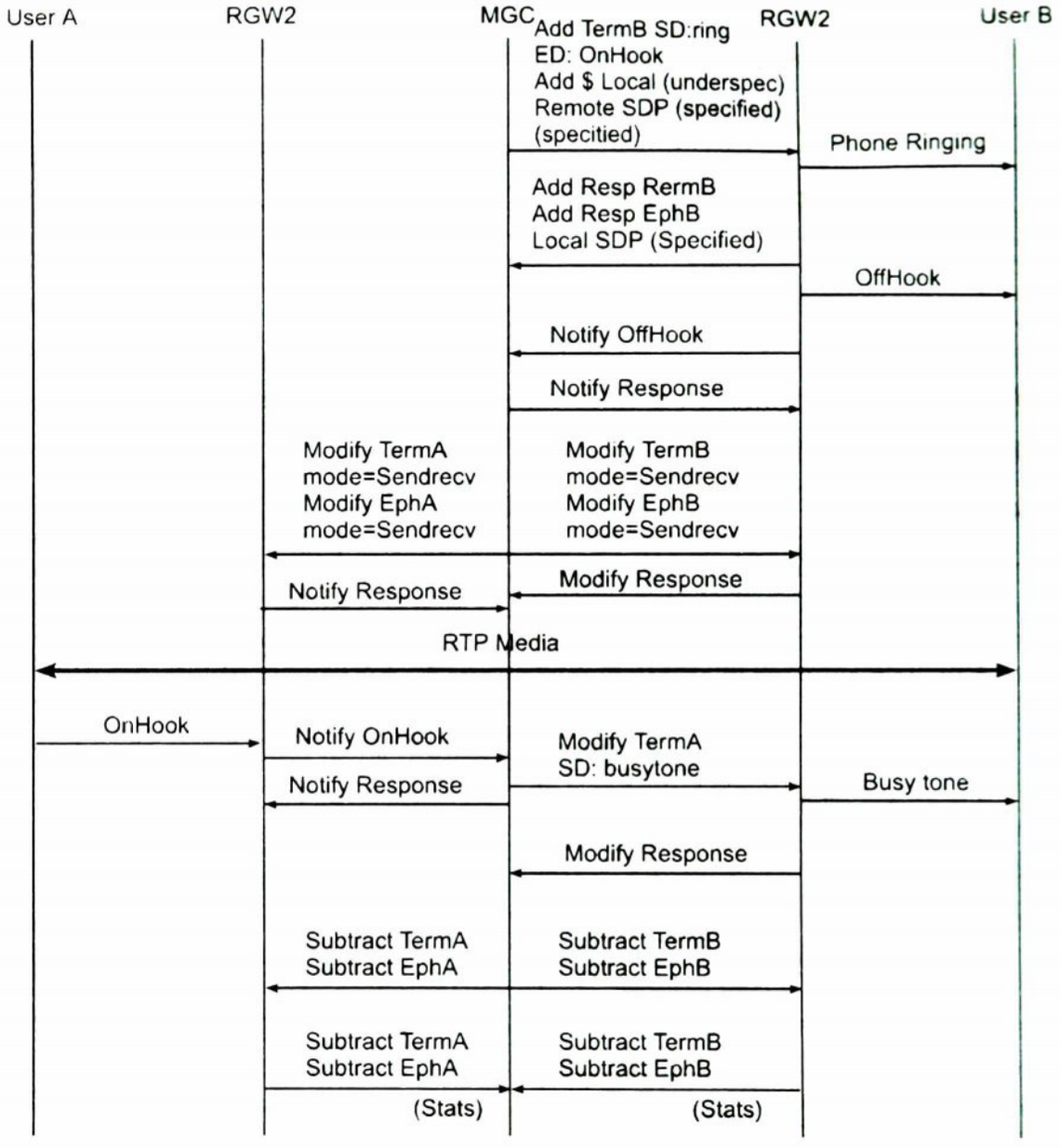
Bước 13: Khi termination B nhấc máy, RGW2 sẽ báo cáo với MGC bằng lệnh "*Notify*".

MG2 to MGC:

MEGACO/1 [207.176.47.89]: 26000



Hình 6.6a: Lưu đồ cuộc gọi giữa hai người dùng thuộc hai RGW được quản lý bởi cùng một MGC.



Hình 6.6b: Lưu đồ cuộc gọi giữa hai người dùng thuộc hai RGW được quản lý bởi cùng một MGC

```
Transaction = 3000 {
  Context = 2 {
    Notify = TermB {ObservedEvents =1234 {
      20000202T10020000:al/of}}
  }
}
```

Bước 14: MGC đáp ứng bằng một bản tin Reply.

```
MGC to MG2:
  MEGACO/1 [216.33.33.61]: 27000
  Reply = 3000 {
     Context = 2 {Notify = TermB}
  }
```

Bước 15: MGC gửi lệnh *"Modify"* để chuyển 2 termination ở RGW2 sang chế độ gửi và nhận. Signal descriptor cũng ngắt báo hiệu chuông ở termination B. Event descriptor chuẩn bị sự kiện đặt máy để chờ.

```
MGC to MG2:
  MEGACO/1 [216.33.33.61]: 27000
  Transaction = 1238 {
    Context = 2 {
      Modify = TermB {
        Signals { } ; to turn off ringing
        Events = 1235 {al/on},
        Media {
     LocalControl {
                Mode = SendRecv,
              }
        }
      }
      Modify = EphB{
        Media {
     LocalControl {
                Mode = SendRecv,
              }
        }
      }
    }
```

Bước 16: RGW2 trả lời bằng bản tin Reply.

```
MG2 to MGC:
  MEGACO/1 [207.176.47.89]: 26000
  Reply = 1238 {
   Context = 2 {Modify = TermB , Modify = EphB}
  }
```

Bước 17: MGC gửi lệnh *"Modify"* tới RGW1 để chuyển chế độ của 2 termination sang chế độ gửi và nhận, ngắt hồi âm chuông ở termination A, thông báo các thông tin về địa chỉ IP, số cổng cho đầu cuối lô-gíc A.

```
MGC to MG1:
  MEGACO/1 [216.33.33.61]: 27000
  Transaction = 1239 {
   Context = 1 {
    Modify = TermA {
      Media {
       LocalControl {
        Mode = sendrecv}
       }.
       }
      Signals { }
    },
    Modify = EphA {
      Media {
       LocalControl {
        Mode = sendrecv}
            Remote {
v=0
c=IN IP4 207.176.47.90
m=audio 40000 RTP/AVP 4
            }
         } ; RTP profile for G723 is 4
      }
```

```
    }
  }
}
```

Bước 18: RGW1 sau khi thực hiện các lệnh trên sẽ gửi Reply cho MGC Lúc này hai đầu cuối có thể trao đổi lưu lượng theo các giao thức RTP/RTCP.

Giả sử người dùng A đặt máy, sự kiện này được RGW1 phát hiện và báo cáo với MGC qua lệnh "*Notify*".

```
MG1 to MGC:
  MEGACO/1 [209.110.59.34]: 25000
  Reply = 1239 {
    Context = 1 {Modify = TermA, Modify = EphA}
  }
```

Bước 19: RGW1 gửi lệnh "*Notify*" cho MGC báo cáo người dùng A đặt máy.

```
RGW1 to MGC:
  MEGACO/1 [209.110.59.34]:25000
  Transaction = 2002 {
    Context = 1 {
      Notify = TermA {ObservedEvents =1112 {
        20010202T10030000:al/on}
      }
    }
  }
```

Bước 20: MGC gửi bản tin Reply cho RGW1.

```
MGC to RGW1:
  MEGACO/1 [216.33.33.619]:27000
  Reply = 2002 {
    Context = 1 {
      Notify = TermA
    }
  }
```

Bước 21: MGC gửi lệnh *"Modify"* tới RGW2 yêu cầu chuyển hai đầu cuối ở context 2 sang chế độ chỉ nhận và gửi âm báo bận tới đầu cuối người dùng A.

```
MGC to RGW2:
  MEGACO/1 [216.33.33.61]: 27000
  Transaction = 1240 {
   Context = 2 {
    Modify = TermB {
     Signals {cg/bt}
      Media {
         LocalControl {
         Mode = recvonly}
          }
    },
    Modify = EphB {
      Media {
         LocalControl {
         Mode = recvonly}
          }
       }
    }
   }
  }
```

Bước 22: RGW2 thực hiện lệnh và gửi bản tin Reply cho MGC.

```
MG2 to MGC:
  MEGACO/1 [207.176.47.89]: 26000
  Reply = 1240 {
    Context = 2 {
      Modify= TermB, Modify = EphB}
  }
```

Bước 23: MGC gửi lệnh "*Subtract*" tới RGW1 yêu cầu xoá 2 termination trong context 1, đồng thời xoá luôn context 1. Các số liệu thống kê mà MGC yêu cầu được chỉ ra trong Audit descriptor.

```
MGC to MG1
  MEGACO/1 [216.33.33.61]: 27000
  Transaction = 1241 {
    Context = 1 {
      Subtract = TermA {Audit{ }},
      Subtract = EphA {Audit{Statistics}}
    }
  }
```

Bước 24: RGW1 thực hiện lệnh và gửi bản tin Reply cho MGC bao gồm các thông tin thống kê về liên kết vừa thiết lập.

```
MG1 to MGC:
  MEGACO/1 [209.110.59.34]:25000
  Reply = 1241 {
    Context = 1 {
      Subtract = TermA
      Subtract = EphA {
        Statistics {
          rtp/ps=1234, ; packets sent
          nt/os=56789, ; octets sent
          rtp/pr=987, ; packets received
          nt/or=65432, ; octets received
          rtp/pl=10, ;  % packets lost
          rtp/jit=30,
          rtp/delay=30 ; average latency
        }
      }
    }
  }
```

Bước 25: Tương tự bước 23, nhưng thực hiện với RGW2.

MGC to MG2:

```
MEGACO/1 [216.33.33.61]: 27000
Transaction = 1242 {
  Context = 2 {
    Subtract = TermB {Audit{ }},
    Subtract = EphB {Audit{Statistics}}
  }
}
```

Bước 26: Tương tự bước 24,nhưng thực hiện với RGW2.

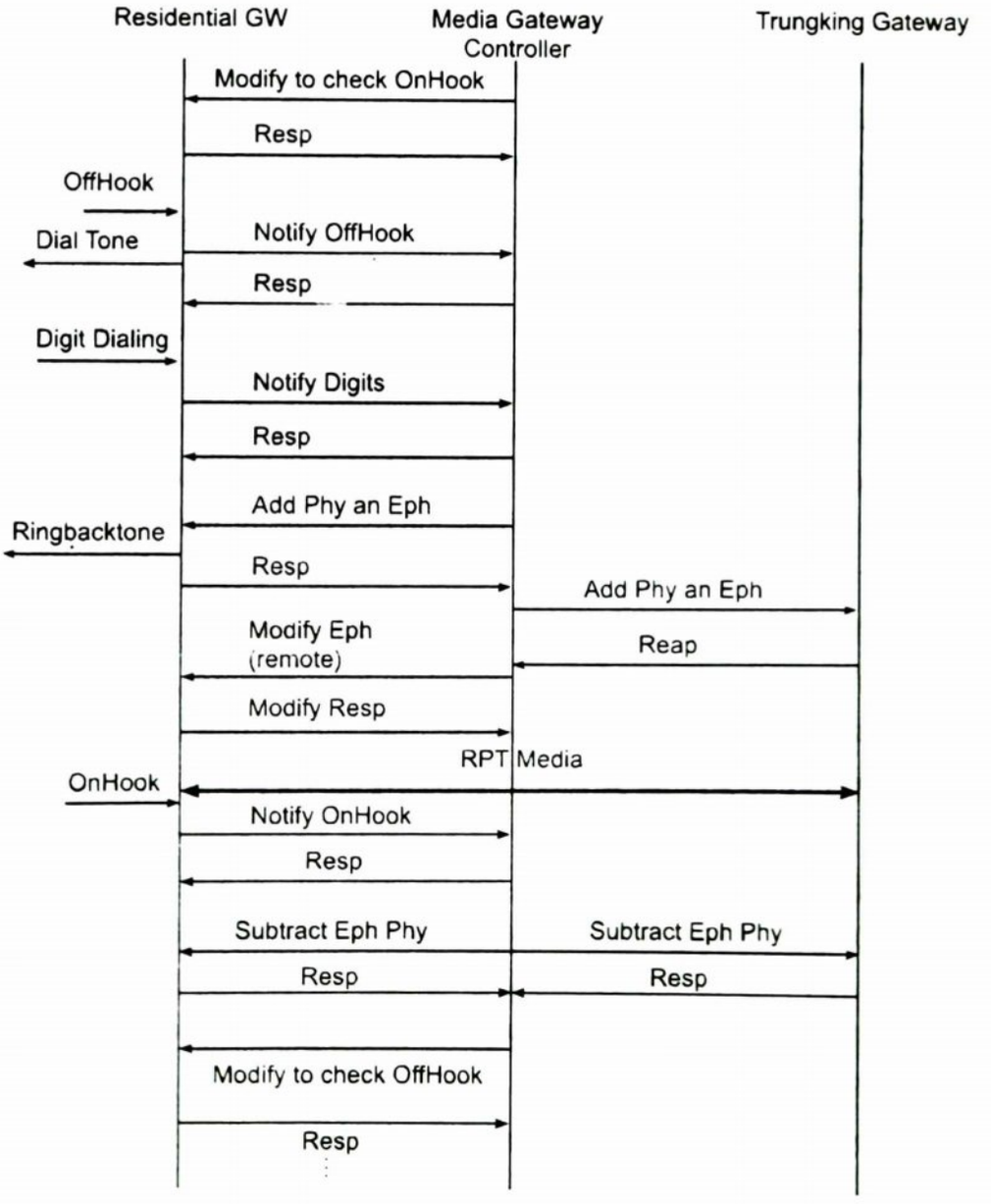
MG2 to MGC:

```
MEGACO/1 [207.176.47.89]:26000
Reply = 1242 {
  Context = 2 {
    Subtract = TermB
      Subtract = EphB {
        Statistics {
          rtp/ps=987, ; packets sent
          nt/os=65432, ; octets sent
          rtp/pr=1234, ; packets received
          nt/or=56789, ; octets received
          rtp/pl=10, ;  % packets lost
          rtp/jit=30,
          rtp/delay=30 ; average latency
        }
      }}}
```

6.6 KẾT NỐI TỪ RGW TỚI TGW

Phần này sẽ mô tả quá trình diễn biến của một cuộc gọi từ RGW tới TGW (Trunking Gateway).

Bước 1: MGC gửi bản tin *"Modify"* tới RGW để phát hiện trạng thái nhấc máy của các kết cuối tại mỗi RGW. Trong bản tin này, có một số nhận dạng báo hiệu và nhận dạng sự kiện được ghi nhớ sẵn (embedded signal descriptor, event descriptor, digit map descriptor). Embedded signal descriptor được sử dụng để gửi âm mời quay số tới đầu cuối nào vừa nhấc máy. Digitmap descriptor chứa mẫu số có thể được quay bởi người dùng. Event descriptor sẽ liệt kê các sự kiện có thể xảy ra. Ngay sau khi RGW phát hiện sự kiện nhấc máy, lệnh "*Modify*" sẽ thiết lập đầu cuối này ở chế độ chỉ nhận.



Hình 6.7: Lưu đồ cuộc gọi từ RGW tới TGW

Bước 2: RGW sau khi nhận được bản tin trên sẽ đáp ứng lại bằng bản tin Reply.

Bước 3: Khi phát hiện đầu cuối offhook, RGW sẽ báo cáo sự kiện này với MGC bằng lệnh "*Notify*".

Bước 4: MGC gửi Reply cho lệnh vừa nhận được.

Digitmap sẽ được kích hoạt ở đầu cuối này, các chữ số được quay sẽ được RGW thu lại.

Bước 5: RGW báo cáo các con số vừa nhận được cho MGC qua lệnh "*Notify*".

Bước 6: MGC gửi bản tin Reply cho RGW.

Bước 7: MGC sẽ ra lệnh cho RGW tạo một đầu cuối lô-gíc (đại diện cho đầu cuối bị gọi), lệnh "*Add*" sẽ đặt đầu cuối vật lý và đầu cuối lô-gíc vào 1 context. Cả hai đầu cuối này đều được thiết lập ở chế độ chỉ nhận.

Bước 8: RGW sau khi nhận lệnh sẽ thực hiện lệnh và trả lời MGC bằng một bản tin Reply. Trong đó có số ID của context vừa tạo và địa chỉ IP, số cổng dành lưu lượng.

Bước 9: Trong ví dụ này, không đề cập tới bất kỳ một phương pháp báo hiệu cụ thể nào bên ngoài TGW, điều này cho phép giảm độ phức tạp của việc mô tả. MGC gửi lệnh "*Add*" yêu cầu TGW tạo một context, chọn một kênh trung kế để phát triển cuộc gọi (đây là một đầu cuối vật lý), tạo một termination lô-gíc và thêm 2 termination này vào context vừa thành lập. Thông tin về địa chỉ IP, số cổng dành cho lưu lượng của RGW cũng được gửi tới TGW.

Bước 10: TGW sẽ thực hiện các lệnh và tiến hành các báo hiệu cần thiết ra bên ngoài TGW. TGW thiết lập một context có ID là 2. Nhận dạng của kênh trung kế bị chiếm giữ cũng sẽ được gửi tới đầu cuối vật lý trong bản tin đáp ứng lệnh "*Add*". TGW cũng gửi các thông tin về địa chỉ IP và số cổng dành cho lưu lượng trong bản tin Reply tới MGC.

Bước 11: Trong ví dụ này không đề cập tới cách mà MGC thu nhận trạng thái của các đầu cuối cuộc gọi. Nhưng khi MGC có được các thông tin đó, MGC sẽ điều khiển để chuyển trạng thái của các đầu cuối về chế độ "gửi và nhận", các thông tin về địa chỉ IP, số cổng dành cho liên lạc cũng sẽ được chuyển tới. Các thông tin điều khiển này được gửi tới RGW qua lệnh *"Modify"*.

Bước 12: RGW sau khi nhận được bản tin trên sẽ đáp ứng lại bằng bản tin Reply.

Bước 13: MGC cũng tạo các lệnh gửi tới TGW để chuyển chế độ của kết cuối tạm thời sang chế độ "gửi và nhận". Do đó cho phép hình thành một kênh thoại ảo giữa các kết cuối tạm thời.

Bước 14: TGW sau khi nhận được lệnh trên sẽ đáp ứng lại bằng bản tin Reply.

Tới đây quá trình diễn biến cuộc gọi xảy ra tương tự như cuộc gọi từ RGW tới RGW.

6.7 KẾT NỐI TỪ SS7-TGW TỚI RGW

Trong trường hợp này giả sử rằng SG-F và TGW-F được cài dặt trong cùng một thiết bị. Công đoạn báo hiệu từ TGW (cổng trung kế) tới MGC được thực hiện bởi giao thức SIGTRAN. MGC sẽ nhận được bản tin IAM thông báo về một cuộc gọi từ đầu cuối thuộc RGW (bản tin này chứa số thuê bao bị gọi). Sau đó MGC trả lời mạng SS7 (sử dụng cùng kênh mà IAM đã tới) qua SG (cổng báo hiệu) bằng bản tin ACM chỉ ra rằng thuê bao bị gọi sẽ sẵn sàng nhận cuộc gọi này.

Bước 1: Sau đó MGC gửi lệnh "*Modify*" tới RGW mà đầu cuối bị gọi trực thuộc, RGW này được yêu cầu gửi tín hiệu chuông tới đầu cuối bị gọi và thiết lập đầu cuối này ở chế độ chỉ nhận. (Bạn đọc quan tâm tới bản tin tương ứng, tham khảo mục 6.6).

Bước 2: RGW sau khi nhận được lệnh trên sẽ trả lời MGC bằng bản tin Reply.

Bước 3: Đầu cuối bị gọi sau khi nghe thấy đổ chuông sẽ nhấc máy trả lời. Sự kiện nhấc máy này được RGW phát hiện và báo cáo tới MGC bằng lệnh "*Notify*".

Bước 4: MGC sau khi nhận lệnh trên sẽ trả lời bằng bản tin Reply.

Tiếp theo, MGC sẽ gửi bản tin ANM tới mạng SS7, chỉ ra trạng thái tiếp nhận cuộc gọi của đầu cuối bị gọi.

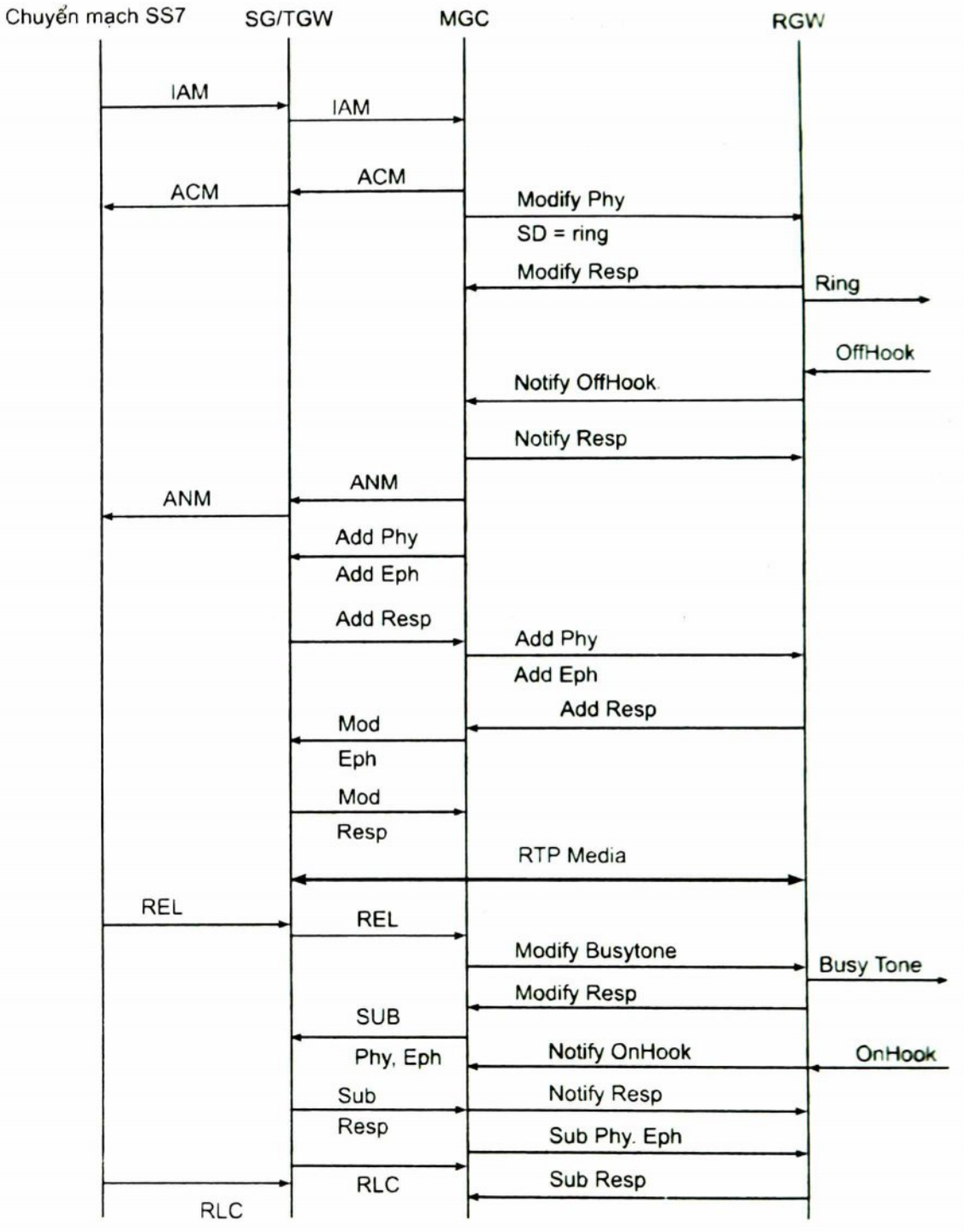
Bước 5: Trong lúc đó MGC cũng gửi lệnh "*Add*" tới TGW để yêu cầu TGW tạo một context với sự có mặt của đầu cuối vật lý đặt ở chế độ "nhận và gửi" (kênh mà chủ gọi chiếm dùng) và một đầu cuối tạm thời đặt ở chế độ chỉ nhận.

Bước 6: Sau khi TGW nhận được lệnh trên sẽ tiến hành thực hiện lệnh và trả lời MGC bằng bản tin Reply.

Trong bản tin này chứa contextID vừa tạo ra với các thông số về địa chỉ IP, số cổng sẽ sử dụng để truyền và nhận lưu lượng.

Bước 7: MGC sau khi nhận được trả lời từ TGW sẽ gửi các thông tin về context, địa chỉ IP, số cổng ở TGW tới RGW. Cũng trong cùng chuyển giao này còn có lệnh tạo đầu cuối tạm thời đặt ở chế độ chỉ nhận và chuyển đầu cuối vật lý bị gọi đặt ở chế độ nhận và gửi vào context (sẽ được tạo ra khi RGW sử dụng lệnh "*Add*").

Bước 8: RGW sau khi nhận được lệnh sẽ tiến hành thực hiện và gửi bản tin Reply tới MGC. Trong chuyển giao này chứa thông tin về context vừa tạo ra và địa chỉ IP cũng như số cổng sử dụng cho phiên liên lạc.



Hình 6.8: Lưu đồ cuộc gọi từ TGW sử dụng SS7 vào RGW

Bước 9: MGC sau khi nhận được các thông tin trên sẽ chuyển chúng cho TGW, đồng thời lệnh "*Modify*" được yêu cầu để chuyển trạng thái của đầu cuối tạm thời ở TGW sang chế độ nhận và gửi.

Bước 10: TGW sau khi nhận và thực hiện lệnh sẽ gửi Reply cho MGC

Các thuộc tính về lưu lượng sẽ được trao đổi giữa 2 GW và đường truyền lưu lượng RTP giữa chúng sẽ được thiết lập.

Hai đầu cuối bắt đầu phiên liên lạc.

Phiên liên lạc sẽ chấm dứt khi một trong 2 đầu cuối đặt máy. Giả sử đầu cuối thuộc TGW sử dụng SS7 sẽ đặt máy trước. Một bản tin REL sẽ được TGW gửi tới MGC. MGC sau khi nhận bản tin này sẽ bắt đầu quá trình giải phóng cuộc gọi.

Bước 11: MGC sẽ yêu cầu RGW gửi âm báo bận tới đầu cuối bị gọi để thông báo cuộc gọi kết thúc. Lệnh *"Modify"* sẽ được gửi đi cùng descriptor mà nó liệt kê sự kiện đặt máy của đầu cuối chủ gọi.

Bước 12: RGW sau khi nhận lệnh sẽ gửi bản tin Reply cho MGC.

Quá trình giải phóng kênh lưu lượng sẽ hoàn thành khi bản tin RLC được gửi tới MGC.

MGC sẽ tiến hành loại bỏ các đầu cuối trong phiên liên lạc.

Bước 13: MGC sẽ gửi lệnh "*Subtract*" tới TGW để loại 2 đầu cuối trong context ở TGW (đồng thời xoá luôn context này), audit descriptor sẽ liệt kê các thông tin thống kê mà MGC yêu cầu.

Bước 14: TGW sẽ đáp ứng lại bằng các số liệu thống kê về phiên liên lạc.

Bước 15: Tương tự như bước 13 nhưng thực hiện với RGW.

Bước 16: Tương tự như bước 14 nhưng thực hiện với TGW.

6.8 MEGACO/H.248 HỖ TRỢ CÁC DỊCH VỤ BỔ SUNG

Dịch vụ chuyển cuộc gọi

Bước 1: Ban đầu MGC gửi lệnh *"Modify"* tới tất cả các RGW để phát hiện sự kiện nhấc máy.

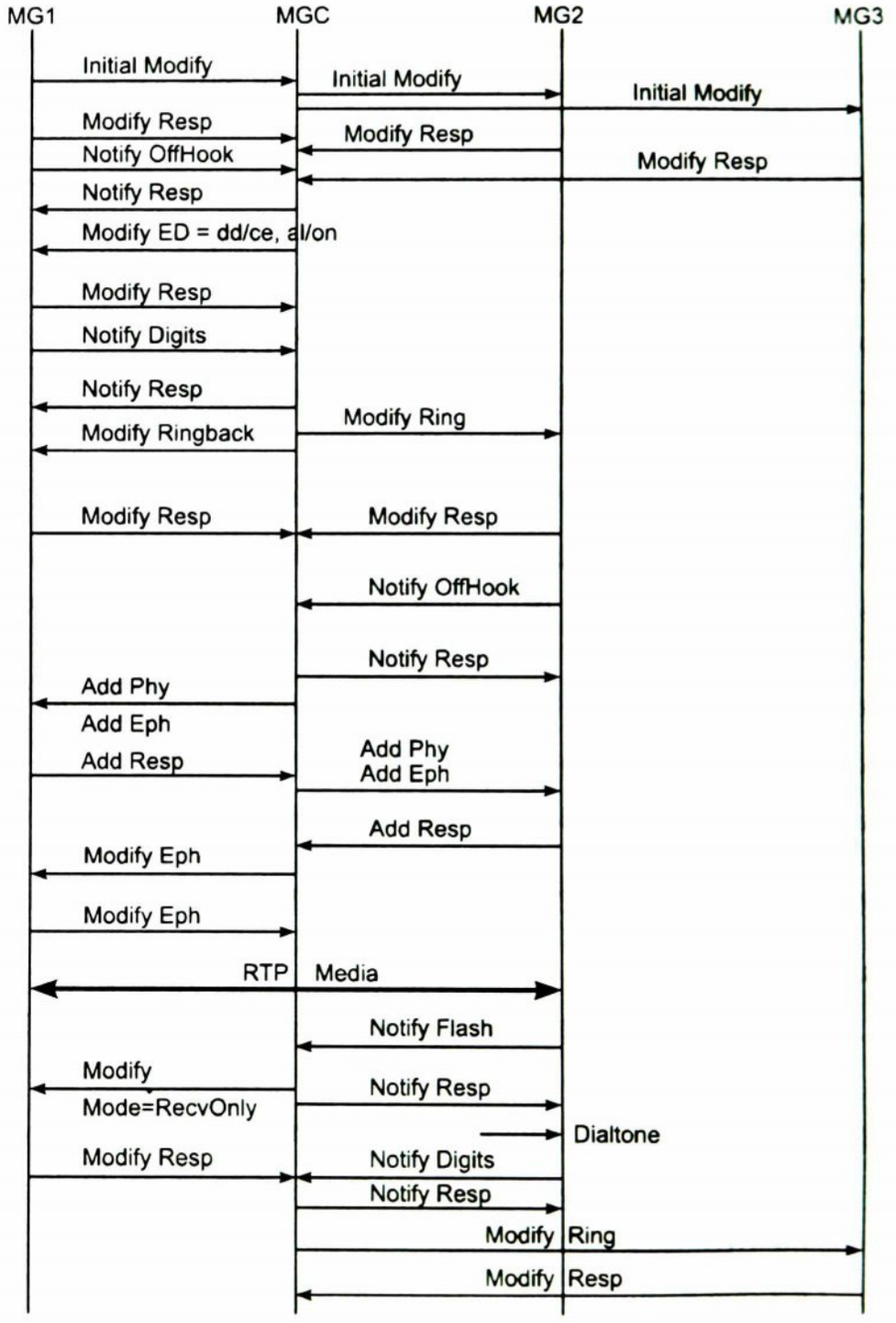
Bước 2: Các RGW lần lượt trả lời lệnh trên của MGC bằng các bản tin Reply

Bước 3: Giả sử người dùng A thuộc RGW1 nhấc máy, sự kiện này sẽ được RGW1 báo cáo tới MGC bằng lệnh "*Notify*".

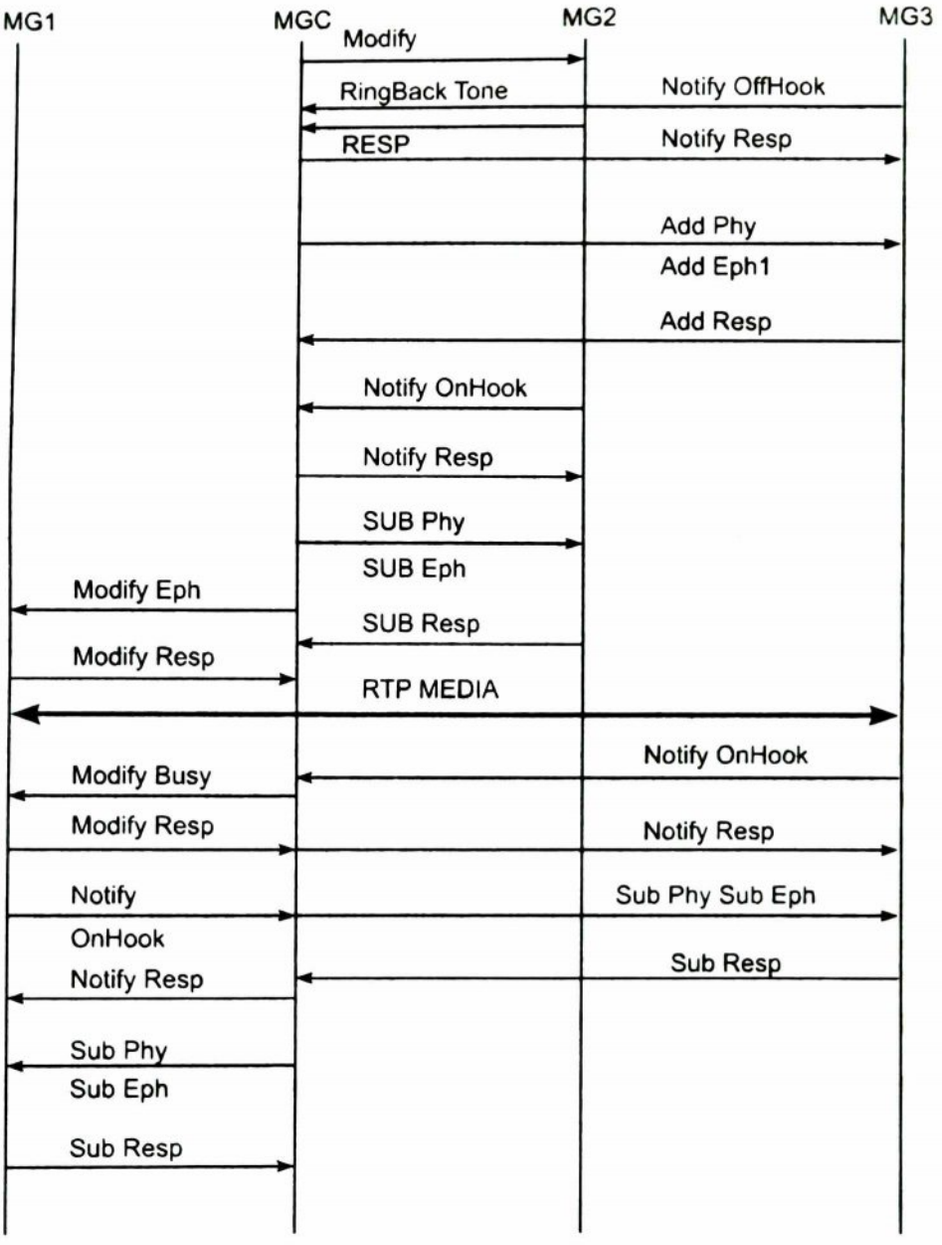
Bước 4: MGC gửi Reply của lệnh này cho RGW1.

Bước 5: MGC sẽ gửi lệnh *"Modify"* tới RGW1, lệnh này gồm 3 descriptor: signal descriptor được sử dụng để gửi âm mời quay số tới người dùng A, digitmap descriptor chứa mô hình mẫu các số có thể quay theo kế hoạch đánh số (ví dụ như 29****), event

descriptor liệt kê các gói DTMF, gói tin hoàn thành quay số và gói tin giám sát trạng thái đặt máy của đầu cuối.



Hình 6.9a: Lưu đồ dịch vụ chuyển cuộc gọi



Hình 6.9b: Lưu đồ dịch vụ chuyển cuộc gọi

Bước 6: RGW1 trả lời MGC bằng một Reply.

RGW1 tiến hành xử lý các descriptor theo thứ tự signal, digitmap, event descriptor. Đầu tiên âm mời quay số sẽ được gửi tới termination A, sau đó digitmap sẽ được cập nhật vào cơ sở dữ liệu của RGW1, digitmap được kích hoạt khi RGW1 thu được sự kiện hoàn thành quay số.

Termination A sau khi nhận được âm mời quay số sẽ tiến hành quay số.

Bước 7: Khi các con số được RGW1 thu đầy đủ và hợp lệ, chúng sẽ được gửi tới MGC bằng lệnh "*Notify*".

Bước 8: MGC xác nhận lệnh trên bằng Reply gửi tới RGW1.

MGC sau khi nhận lệnh trên sẽ phân tích số bị gọi và biết đầu cuối termination B đó thuộc RGW2. MGC tiếp tục điều khiển RGW1.

Bước 9: MGC sẽ gửi cho RGW1 hai lệnh. Lệnh "*Add*" để tạo một context và thêm ngay termination A vào đó. MGC biết rằng termination B rỗi, nó sẽ gửi ringback cho termination A. Lệnh thứ 2 để tạo một đầu cuối lô-gíc A và thêm đầu cuối này vào context vừa tạo ra.

Bước 10: RGW1 sẽ gửi Reply cho MGC bao gồm contextID(1), địa chỉ IP và số cổng dành cho lưu lượng.

Bước 11: MGC sẽ gửi tới RGW2 2 lệnh. Lệnh 1 để tạo một context và thêm termination B vào context này. Báo hiệu chuông cũng được gửi tới termination B nhờ signal descriptor. Lệnh thứ hai sẽ tạo một đầu cuối lô-gíc B và thêm đầu cuối này vào context vừa tạo ra. Các thông tin địa chỉ IP, số cổng của termination A cũng được gửi tới RGW2.

Bước 12: RGW2 sau khi nhận lệnh sẽ thực hiện lệnh và gửi kết quả thực hiện tới MGC , bao gồm contextID(2), địa chỉ IP và số cổng dành cho lưu lượng.

MGC đợi cho termination B nhấc máy.

Bước 13: Khi termination B nhấc máy, RGW2 sẽ báo cáo với MGC bằng lệnh Notify.

Bước 14: MGC đáp ứng bằng một Reply.

Bước 15: MGC gửi lệnh "*Modify*" để chuyển 2 termination ở RGW2 sang chế độ gửi và nhận. Signal descriptor cũng ngắt báo hiệu chuông ở termination B. Event descriptor chuẩn bị sự kiện đặt máy để chờ.

Bước 16: RGW2 trả lời bằng Reply.

Bước 17: MGC gửi lệnh *"Modify"* tới RGW1 để chuyển chế độ của 2 termination sang chế độ gửi và nhận, ngắt hồi âm chuông ở termination A, thông báo các thông tin về địa chỉ IP, số cổng cho đầu cuối lô-gíc A.

Bước 18: RGW1 sau khi thực hiện các lệnh trên sẽ gửi Reply cho MGC

Lúc này hai đầu cuối có thể trao đổi lưu lượng theo các giao thức RTP/RTCP.

Bước 19: Người dùng B ấn phím Flash để quay số tới người dùng C. Sự kiện nhấn phím Flash được RGW2 phát hiện và sẽ gửi báo cáo tới MGC bằng bản tin Reply.

Bước 20: MGC gửi bản tin Reply cho RGW2.

RGW2 sẽ gửi âm mời quay số tới người dùng B, người dùng B bắt đầu quay số.

Bước 21: Khi sự kiện hoàn thành quay số được phát hiện, RGW2 sẽ gửi số bị quay tới MGC bằng lệnh "*Notify*".

Bước 22: MGC trả lời bằng một Reply.

Bước 23: MGC sẽ gửi 2 lệnh "*Add*" tới RGW3 (quản lý termination C). Lệnh 1 yêu cầu thiết lập 1 context và đặt termination C vào context đó. Signal descriptor sẽ tạo chuông báo tới termination C. Lệnh thứ 2 yêu cầu tạo một đầu cuối lô-gíc C, và thêm nó vào context vừa thành lập. Các thông tin địa chỉ IP và số cổng mà termination B sử dụng cũng được gửi tới RGW3.

Bước 24: RGW3 sau khi nhận lệnh và thực hiện lệnh sẽ gửi kết quả thực hiện tới MGC, bao gồm contextID (3), địa chỉ IP và số cổng dành cho lưu lượng.

Bước 25: MGC cũng gửi âm báo trở lại tới termination B bằng signal descriptor trong lệnh *"Modify"* (tới RGW2).

Bước 26: RGW2 gửi Reply cho MGC.

Bước 27: Termination C sau khi nghe chuông reo sẽ nhấc máy và sự kiện này được RGW3 báo cáo tới MGC trong lệnh "*Notify*".

Bước 28: MGC gửi Reply cho RGW3.

Bước 29: MGC gửi lệnh "*Modify*" tới RGW3 để chuyển 2 termination sang chế độ gửi và nhận, đồng thời ngắt chuông ở termination C.

Bước 30: RGW3 gửi Reply cho MGC.

Bước 31: MGC gửi lệnh "*Modify*" tới RGW2 để thiết lập các đầu cuối ở đó sang chế độ nhận và gửi, đồng thời các thông tin về địa chỉ IP, số cổng của termination C cũng được gửi tới cho đầu cuối lô-gíc B.

Bước 32: RGW2 trả lời bằng một Reply.

Lúc này termination A đang đặt ở chế độ chỉ nhận. Termination B và termination C có thể trao đổi lưu lượng. Lúc này người dùng B sẽ đặt máy, chỉ ra hội thoại với người dùng A (chuyển cuộc gọi).

Bước 33: RGW2 sẽ thông báo sự kiện nhấc máy của termination B tới MGC bằng lệnh "*Notify*".

Bước 34: MGC đáp lại bằng Reply tới RGW2.

Bước 35: MGC gửi lệnh "*Subtract*" tới RGW2 yêu cầu xoá 2 termination ở context2 (đồng thời xoá luôn context2).

Bước 36: RGW2 nhận lệnh, thực hiện lệnh và gửi Reply cho MGC (gồm các số liệu thống kê về kết nối).

Người dùng B chính thức được giải phóng và không ở trong liên kết với bất kỳ người dùng nào (A hay C).

Bước 37: MGC gửi lệnh *"Modify"* tới RGW1 yêu cầu chuyển chế độ của 2 termination trong context1 sang chế độ gửi và nhận. Các thông tin về địa chỉ IP và số cổng dành cho lưu lượng của termination C cũng được gửi đi.

Bước 38: RGW1 thực hiện lệnh và gửi Reply cho MGC.

Bước 39: Tương tự bước 37 nhưng thực hiện với RGW3.

Bước 40: Tương tự bước 38 nhưng thực hiện với RGW3.

Tới đây người dùng A và C thực hiện trao đổi lưu lượng.

Quá trình giải phóng kết nối và các đầu cuối được thực hiện như trong các dịch vụ khác.

Dịch vụ chờ cuộc gọi

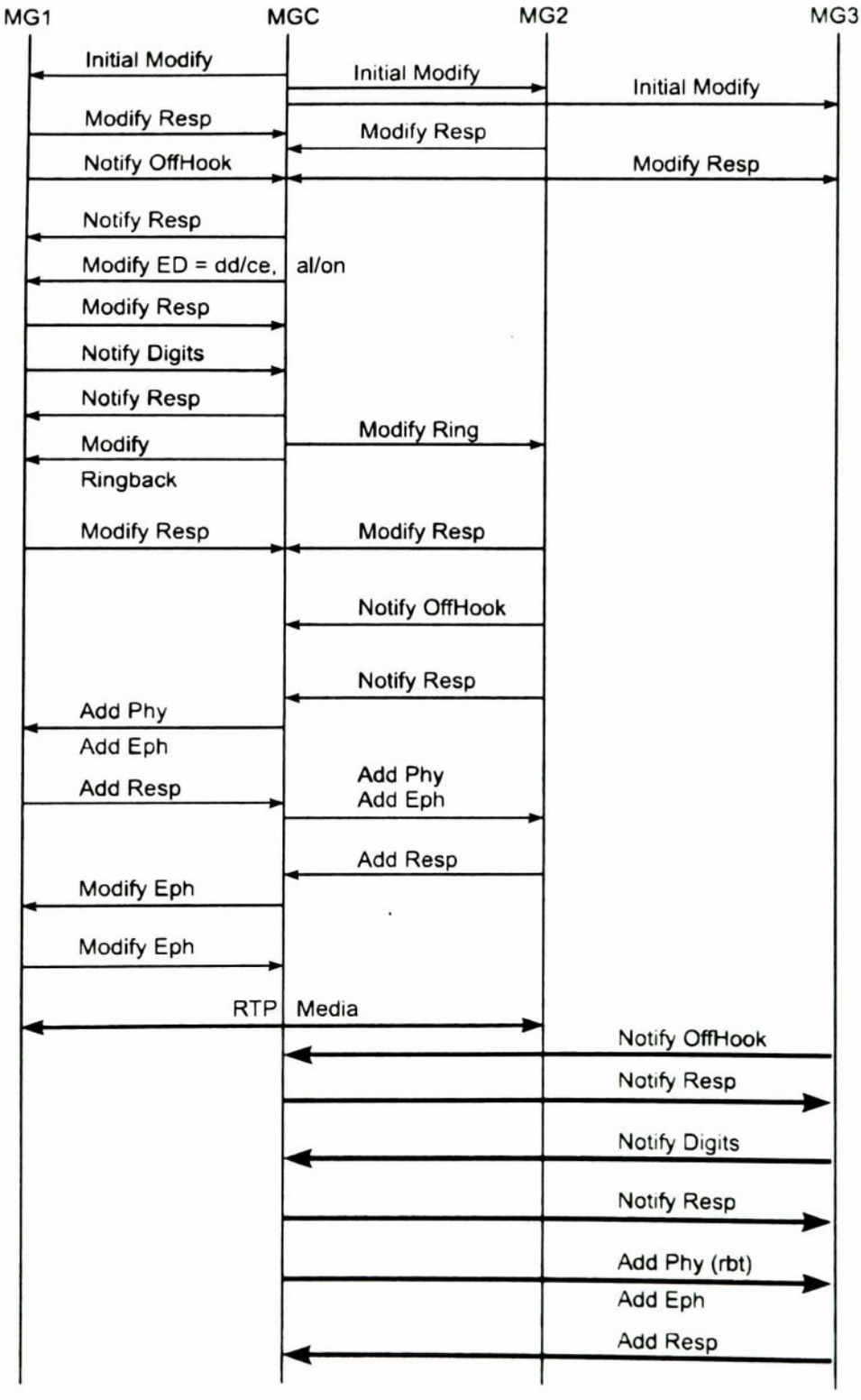
Dịch vụ chờ cuộc gọi cho phép một người dùng có thể nhận hai cuộc gọi cùng lúc bằng cách chuyển tiếp giữa hai cuộc gọi. Có thể mô tả chi tiết như sau: Người dùng A gọi người dùng B, cuộc gọi này đang diễn ra thì người dùng C gọi người dùng B. Người dùng B sẽ nhận được âm báo chờ cuộc gọi và chuyển sang đàm thoại với người dùng C (qua việc ấn phím Flash). Khi cuộc gọi này kết thúc (người dùng C đặt máy), người dùng B quay lại cuộc gọi đầu tiên với người dùng A.

Phần này sẽ mô tả tiến trình xử lý cuộc gọi đối với dịch vụ chờ cuộc gọi giữa 3 termination được quản lý bởi cùng một MGC.

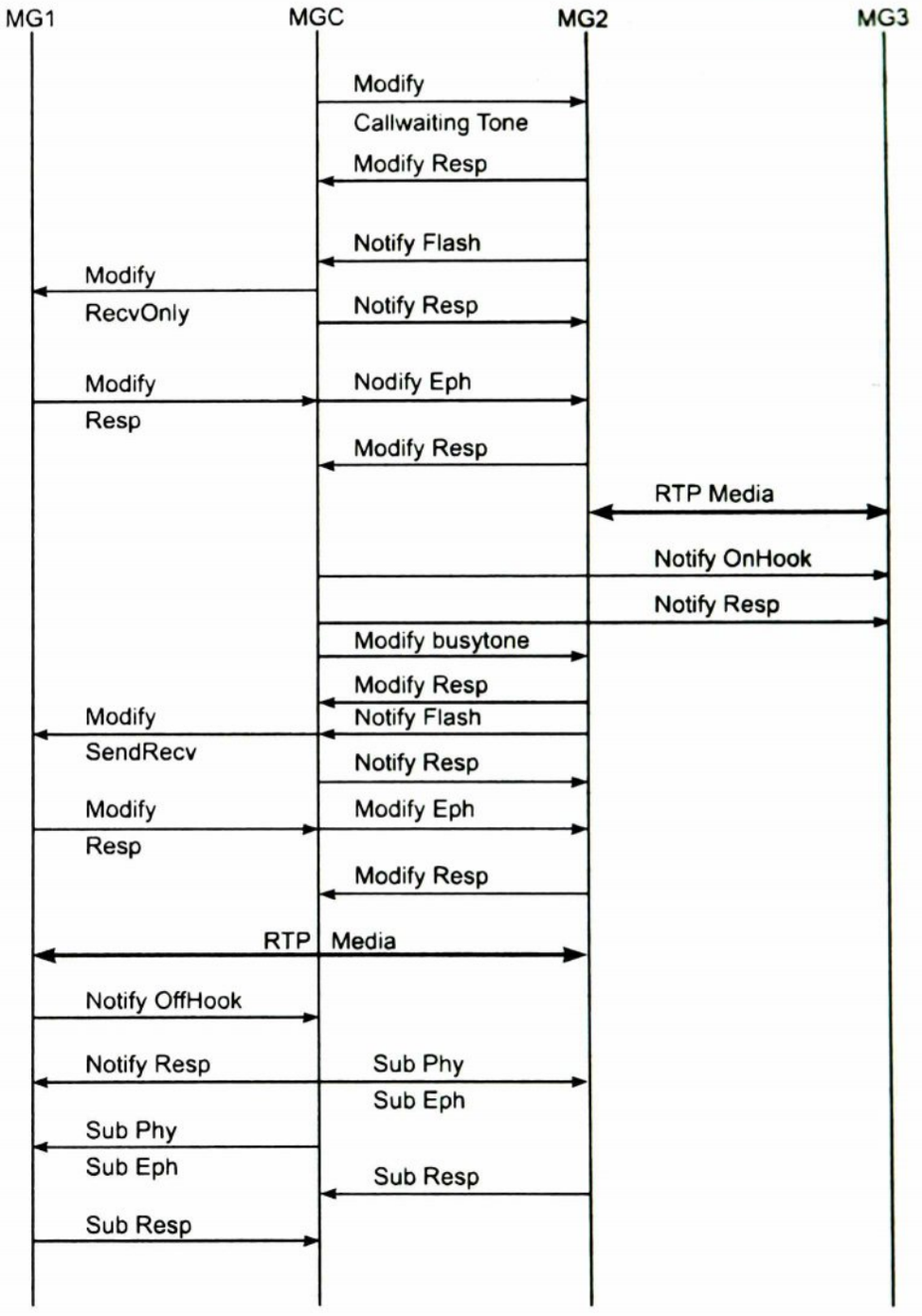
Bước 1: Ban đầu MGC gửi lệnh "*Modify*" tới tất cả các RGW để phát hiện sự kiện nhấc máy.

Bước 2: Các RGW lần lượt trả lời lệnh trên của MGC bằng các Reply

Bước 3: Giả sử người dùng A thuộc RGW1 nhấc máy, sự kiện này sẽ được RGW1 báo cáo tới MGC bằng lệnh "*Notify*".



Hình 6.10a: Lưu đồ dịch vụ chờ cuộc gọi



Hình 6.10b: Lưu đồ dịch vụ chờ cuộc gọi

Bước 4: MGC gửi Reply của lệnh này cho RGW1.

Bước 5: MGC sẽ gửi lệnh "*Modify*" tới RGW1, lệnh này gồm 3 descriptor: signal descriptor được sử dụng để gửi âm mời quay số tới người dùng A, digitmap descriptor chứa mô hình mẫu các số có thể quay theo kế hoạch đánh số (ví dụ như 29****), event descriptor liệt kê các gói DTMF, gói tin hoàn thành quay số và gói tin giám sát trạng thái đặt máy của đầu cuối.

Bước 6: RGW1 trả lời MGC bằng một Reply.

RGW1 tiến hành xử lý các descriptor theo thứ tự signal, digitmap, event descriptor. Đầu tiên âm mời quay số sẽ được gửi tới termination A, sau đó digitmap sẽ được cập nhật vào cơ sở dữ liệu của RGW1, digitmap được kích hoạt khi RGW1 thu được sự kiện hoàn thành quay số.

Termination A sau khi nhận được âm mời quay số sẽ tiến hành quay số.

Bước 7: Khi các con số được RGW1 thu đầy đủ và hợp lệ, chúng sẽ được gửi tới MGC bằng lệnh "*Notify*".

Bước 8: MGC xác nhận lệnh trên bằng Reply gửi tới RGW1.

MGC sau khi nhận lệnh trên sẽ phân tích số bị gọi và biết đầu cuối termination B đó thuộc RGW2. MGC tiếp tục điều khiển RGW1.

Bước 9: MGC sẽ gửi cho RGW1 hai lệnh. Lệnh "*Add*" để tạo một context và thêm ngay termination A vào đó. MGC biết rằng termination B rỗi, nó sẽ gửi hồi âm chuông cho termination A. Lệnh thứ 2 để tạo một đầu cuối lô-gíc A và thêm đầu cuối này vào context vừa tạo ra.

Bước 10: RGW1 sẽ gửi bản tin Reply cho MGC bao gồm contextID(1), địa chỉ IP và số cổng dành cho lưu lượng.

Bước 11: MGC sẽ gửi tới RGW2 2 lệnh. Lệnh 1 để tạo một context và thêm termination B vào context này. Báo hiệu chuông cũng được gửi tới termination B nhờ signal descriptor. Lệnh thứ hai sẽ tạo một đầu cuối lô-gíc B và thêm đầu cuối này vào context vừa tạo ra. Các thông tin địa chỉ IP, số cổng của termination A cũng được gửi tới RGW2.

Bước 12: RGW2 sau khi nhận lệnh sẽ thực hiện lệnh và gửi kết quả thực hiện tới MGC, bao gồm contextID(2), địa chỉ IP và số cổng dành cho lưu lượng.

MGC đợi cho termination B nhấc máy.

Bước 13: Khi termination B nhấc máy, RGW2 sẽ báo cáo với MGC bằng lệnh "*Notify*".

Bước 14: MGC đáp ứng bằng một Reply.

Bước 15: MGC gửi lệnh *"Modify"* để chuyển 2 termination ở RGW2 sang chế độ gửi và nhận. Signal descriptor cũng ngắt báo hiệu chuông ở termination B. Event descriptor chuẩn bị sự kiện đặt máy để chờ.

Bước 16: RGW2 trả lời bằng Reply.

Bước 17: MGC gửi lệnh "*Modify*" tới RGW1 để chuyển chế độ của 2 termination sang chế độ gửi và nhận, ngắt hồi amm chuông ở termination A, thông báo các thông tin về địa chỉ IP, số cổng cho đầu cuối lô-gíc A.

Bước 18: RGW1 sau khi thực hiện các lệnh trên sẽ gửi Reply cho MGC

Lúc này hai đầu cuối có thể trao đổi lưu lượng theo các giao thức RTP/RTCP.

Khi cuộc gọi trên đang diễn ra, người dùng C nhấc máy và quay số.

Bước 19: Sự kiện trên được RGW3 phát hiện và gửi báo cáo bằng lệnh "*Notify*" tới MGC.

Bước 20: MGC gửi Reply của lệnh này cho RGW3.

Bước 21: MGC sẽ gửi lệnh "*Modify*" tới RGW3, lệnh này gồm 3 descriptor: signal descriptor được sử dụng để gửi âm mời quay số tới người dùng C, digitmap descriptor chứa mô hình mẫu các số có thể quay theo kế hoạch đánh số (ví dụ như 29****), event descriptor liệt kê các gói DTMF, gói tin hoàn thành quay số và gói tin giám sát trạng thái onhook của đầu cuối.

Bước 22: RGW3 trả lời MGC bằng một Reply.

RGW3 tiến hành xử lý các descriptor theo thứ tự signal, digitmap, event descriptor. Đầu tiên âm mời quay số sẽ được gửi tới termination C, sau đó digitmap sẽ được cập nhật vào cơ sở dữ liệu của RGW3, digitmap được kích hoạt khi RGW3 thu được sự kiện hoàn thành quay số.

Termination C sau khi nhận được âm mời quay số sẽ tiến hành quay số.

Bước 23: Khi các con số được RGW3 thu đầy đủ và hợp lệ, chúng sẽ được gửi tới MGC bằng lệnh "*Notify*".

Bước 24: MGC xác nhận lệnh trên bằng Reply gửi tới RGW1.

Bước 25: MGC sau khi phân tích số bị gọi sẽ nhận ra có một cuộc gọi khác đang chờ người dùng B. Trước khi gửi các lệnh tới RGW2, MGC gửi lệnh "*Add*" tới RGW3 yêu cầu tạo lập một context, một đầu cuối lô-gíc C cũng được tạo ra. Cả hai đầu cuối (vật lý và lô-gíc) đều được thêm vào context vừa tạo. Cả hai đầu cuối này được đặt ở chế độ chỉ nhận. Các thông tin về địa chỉ IP, số cổng sử dụng của termination B cũng được gửi tới RGW3.

Bước 26: RGW3 thực hiện lệnh trên và gửi Reply cho MGC bao gồm số ID của context(3) vừa tạo lập, địa chỉ IP, số cổng sẽ dành để liên lạc.

Bước 27: MGC gửi lệnh *"Modify"* tới RGW2 với âm báo đợi cuộc gọi trong descriptor signal.

Bước 28: RGW2 thực hiện lệnh trên và gửi Reply cho MGC.

Ngươi dùng B sau khi nghe thấy âm báo mời quay số sẽ ấn phím Flash.

Bước 29: Sự kiện ấn phím Flash được RGW2 phát hiện và báo cáo về MGC bằng lệnh "*Notify*".

Bước 30: MGC gửi Reply cho RGW2.

Bước 31: MGC gửi lệnh *"Modify"* cho RGW2, các thông tin về địa chỉ IP, số cổng của termination C cũng được gửi tới RGW2.

Bước 32: Signal descriptor sẽ ngắt âm báo đợi cuộc gọi, các thông tin về địa chỉ vừa nhận sẽ được đầu cuối lô-gíc B cập nhật. RGW2 gửi Reply cho MGC.

Bước 33: MGC gửi lệnh "*Modify*" cho RGW3 yêu cầu ngắt hồi âm chuông và chuyển cả 2 đầu cuối trong context3 sang chế độ nhận và gửi.

Bước 34: RGW3 thực hiện lệnh và gửi bản tin Reply cho MGC.

Tới lúc này người dùng C và B có thể đàm thoại dưới sự điều khiển của giao thức RTP/RTCP.

Giả sử người dùng C đặt máy.

Bước 35: Sự kiện trên được RGW3 phát hiện và gửi báo cáo tới MGC bằng lệnh "*Notify*".

Bước 36: MGC gửi Reply cho RGW3.

Bước 37: MGC gửi lệnh *"Modify"* tới RGW2 yêu cầu gửi âm báo bận tới đầu cuối người dùng B, sau đó chuyển cả hai đầu cuối trong context 2 sang chế độ chỉ nhận.

Bước 38: RGW2 thực hiện lệnh và gửi Reply cho MGC.

Người dùng B sẽ bấm phím Flash để tiếp tục cuộc gọi với người dùng A.

Bước 39: Sự kiện trên được RGW2 phát hiện và sẽ gửi Reply tới MGC bằng lệnh "*Notify*".

Bước 40: MGC gửi Reply cho RGW2.

Bước 41: MGC gửi lệnh "*Modify*" tới RGW2 yêu cầu chuyển 2 termination của context 2 sang chế độ nhận và gửi, đồng thời gửi các thông tin về địa chỉ và số cổng của termination A.

Bước 42: RGW2 thực hiện lệnh (cập nhật các thông tin về địa chỉ cho đầu cuối lô-gíc A, ngắt âm báo bận ở termination B) và gửi Reply cho MGC.

Người dùng A và B lại tiếp tục cuộc gọi.

Giả sử người dùng A đặt máy.

Bước 43: Sự kiện đặt máy trên được RGW1 phát hiện và gửi báo cáo qua lệnh "*Notify*" tới MGC.

Bước 44: MGC gửi Reply cho RGW1.

Bước 45: MGC gửi lệnh "*Modify*" cho RGW2 yêu cầu gửi âm báo bận cho termination B, đồng thời chuyển chế độ của 2 đầu cuối sang chế độ chỉ nhận.

Bước 46: RGW2 thực hiện lệnh và gửi bản tin Reply cho MGC.

Bước 47: MGC gửi lệnh "*Subtract*" cho RGW2 yêu cầu xoá 2 termination khỏi context2 (đồng thời xóa luôn cả context2).

Bước 48: RGW2 thực hiện lệnh và gửi bản tin Reply gồm các dữ liệu thống kê về cuộc gọi (đã định trước) cho MGC.

Bước 49: Như bước 47, nhưng tiến hành với RGW1.

Bước 50: Như bước 48, nhưng tiến hành với RGW1.

Chương 7

GIAO THỨC TRUYỀN TẢI BÁO HIỆU SIGTRAN

7.1 MÔ HÌNH KIẾN TRÚC MẠNG TÍCH HỢP

Công nghiệp truyền thông đang trải qua thời kì có nhiều thay đổi lớn, hướng tới sự hội tụ công nghệ và tích hợp các dịch vụ. Dữ liệu đang ngày càng chiếm tỉ trọng lưu lượng lớn so với thoại. Các nhà điều hành mạng đang tìm cách để hợp nhất thoại và dữ liệu, các nền tảng và các dịch vụ để giảm chi phí vận hành, bảo dưỡng và chi phí ban đầu của mạng. Trong số các giải pháp công nghệ đưa ra thì công nghệ IP (Internet Protocol) được xác định là công nghệ truyền thông nền tảng trong việc xây dựng các mạng hội tụ và dịch vụ tích hợp mới. Xu thế hội tụ mạng chuyển mạch kênh và mạng IP đang diễn ra mạnh mẽ và nhanh chóng. Các nhà điều hành mạng điện thoại cố định và di động đang thiết kế mạng IP toàn phần, bao gồm cả việc hỗ trợ cho các giao thức báo hiệu của hệ thống báo hiệu SS7. IP cung cấp một phương pháp hiệu quả để truyền dữ liệu của người dùng và cho phép các nhà điều hành mở rộng các mạng của họ, cũng như cung cấp các dịch vụ mới. Các dịch vụ thông tin phổ biến bao gồm cả các dịch vụ bản tin ngắn SMS (Short Message Service) đã góp phần thúc đẩy sự phát triển nhanh chóng của các mạng báo hiệu. Theo đó thì các mạng có quy mô và tính mềm dẻo lớn như mạng Internet và các công nghệ của nó là thực sự cần thiết. So sánh với mạng chuyển mạch kênh TDM truyền thống mạng, IP có một số ưu điểm vượt trội sau:

- **Dễ dàng triển khai:** khi sử dụng các cổng báo hiệu SG (chẳng hạn như nhóm truy cập dịch vụ ASG (Access Service Group), không cần thiết phải loại bỏ mạng báo hiệu SS7 hiện có và quá trình nâng cấp mạng trong tương lai là hoàn toàn trong suốt.

- **Giảm chi phí đầu tư cơ bản:** không cần phải đầu tư nhiều cho các thành phần báo hiệu đã có trước đó.

- **Hiệu quả hoạt động cao:** truyền tải báo hiệu qua mạng IP không yêu cầu các luồng tín hiệu E1/T1 qua các vòng ring SDH. Bằng cách sử dụng các công nghệ mới như "IP over SDH" (IP qua SDH) và "IP over fiber" (IP qua cáp sợi quang) chẳng hạn, có thể đạt được thông lượng cao hơn nhiều.

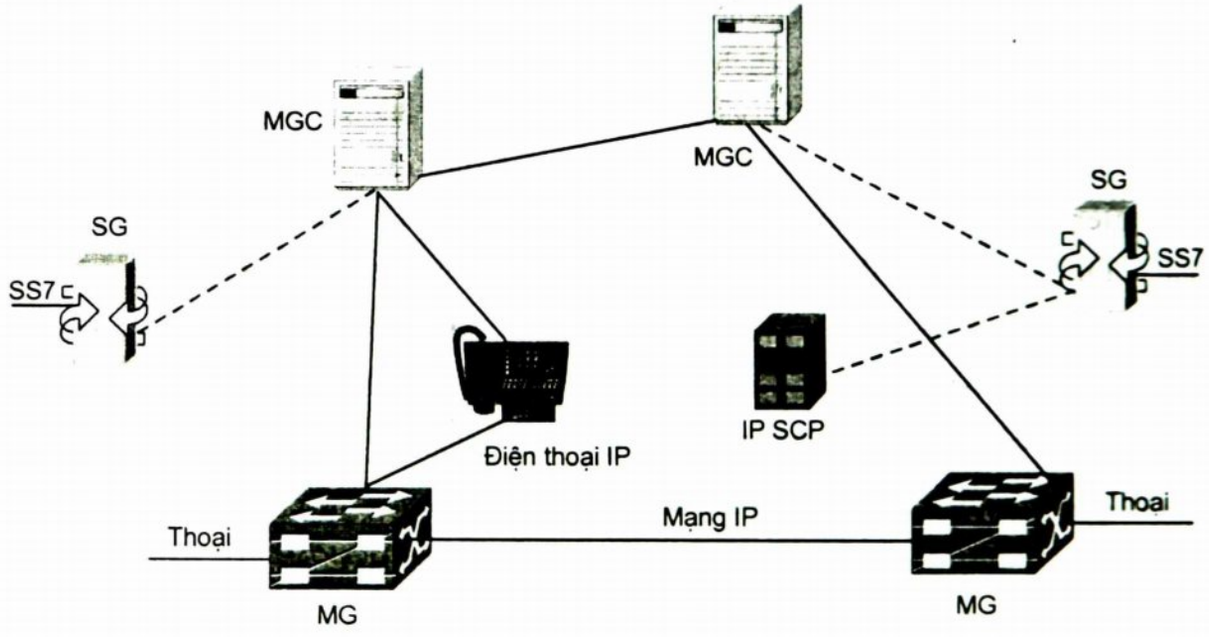
- Băng thông lớn hơn: thông tin của SIGTRAN qua mạng IP không bắt buộc phải chứa các tuyến giống như ở trong hệ thống mạng báo hiệu SS7. Mạng IP mềm dẻo hơn nhiều so với mạng TDM truyền thống.

- Cải thiện dịch vụ: mạng IP tạo ra một loạt các giải pháp và các dịch vụ giá trị gia tăng VAS (Value Added Services) mới.

Với các ưu điểm kể trên của mạng IP thì vấn đề tích hợp giữa IP và mạng PSTN đang trở nên hết sức hấp dẫn và cấp thiết. Một số nhóm chuyên gia đã tiến hành nghiên cứu về lĩnh vực này theo hình thức hợp tác hoặc độc lập, bao gồm:

- ETSI (dự án TIPHON)
- ITU (nhóm làm việc H.323)
- Softswitch Consortium (Hiệp hội chuyển mạch mềm)
- Nhóm làm việc IETF-MGCP

Các nhóm này có nhiệm vụ thiết lập một mô hình kiến trúc chung, định nghĩa giao diện giữa mạng PSTN và mạng IP nhằm hỗ trợ thoại (hoặc bất cứ luồng phương tiện nào khác) qua mạng IP. Một vài loại thiết bị nút mạng mới đã được giới thiệu, chức năng và nhiệm vụ của chúng cũng đã được định nghĩa. Hình 7.1 minh họa một mô hình kiến trúc mạng tích hợp.



Hình 7.1: Kiến trúc mạng tích hợp

Trên hình 7.1 ta thấy có các thành phần mạng sau:

• MGC (Media Gateway Controller): có nhiệm vụ xử lý và điều khiển cuộc gọi, điều khiển các SG và MG liên quan và điều khiển việc truy nhập đến/từ mạng PSTN.

• SG (Signaling Gateway): giao tiếp với mạng SS7 và truyền các bản tin báo hiệu đến các nút IP.

• MG (Media Gateway): Theo sự điều khiển của MGC, MG thực hiện nhiệm vụ chuyển mạch, chuyển đổi và đóng gói lưu lượng thoại và truyền chúng đến đích.

• IP SCP (IP Service Control Point): Điểm điều khiển dịch vụ IP. Điểm điều khiển dịch vụ này nằm hoàn toàn trong mạng IP nhưng địa chỉ của nó có thể từ mạng SS7.

• IP Phone: thiết bị đầu cuối IP.

Các thuật ngữ này thường rất ít gặp. Trong môi trường H.323, MGC được xem như là "Gatekeeper" trong khi đó SG và MG được biết đến như là "Gateway". Thông thường cả ba thành phần này được gọi chung trong thuật ngữ "Softswitch".

Cần chú ý rằng MG, SG và MGC là các thực thể lô-gíc. Về mặt thực hiện vật lý, các chức năng có thể được kết hợp trong các nút mạng.

Ví dụ:

Ở các mạng PSTN của Bắc Mỹ, các kênh báo hiệu và truyền thông được truyền trên các đường vật lý tách biệt. Do vậy nó tạo ra sự phân biệt về mặt vật lý giữa các nút SG và MG.

Còn đối với các mạng PSTN ở châu Âu, thông thường thì các kênh báo hiệu và thoại cùng nhau chia sẻ các đường truyền vật lý. Điều này khiến cho ta cảm giác là SG và MG được đặt cùng vị trí với nhau.

Mô hình kiến trúc này cho phép các nhà cung cấp thiết bị mạng xây dựng các thiết bị cổng nối và bộ điều khiển mà có thể được triển khai trong các mạng thực tế. Còn với các nhà điều hành mạng, họ có thể xây dựng các mạng thực sự tích hợp giữa PSTN và IP, qua đó cung cấp các dịch vụ hấp dẫn cho khách hàng.

Thêm vào đó để định nghĩa các nút mạng - sử dụng để liên kết mạng PSTN và IP, các nhóm làm việc nêu trên đã đưa ra các giao thức thích hợp để liên kết giữa chúng trong mạng chung. Có hai bộ các tiêu chuẩn được biết đến rộng rãi đó là:

• H.323 của ITU (đã được đề cập đến trong chương 4)

• Mô hình của IETF - bao gồm SIP, SIGTRAN, RTP và MGCP (hoặc MEGACO).

IETF thông qua một số nhóm làm việc đã xây dựng một mô hình tương tự H.323. Các nút mạng được định nghĩa trong mô hình này đã được chỉ ra trong hình 7.1. Các giao thức sử dụng trong mô hình này gồm có:

- SIGTRAN (gồm SCTP và các lớp thích ứng): đây là bộ giao thức được sử dụng để truyền tải báo hiệu SS7 giữa SG và MGC.

- MEGACO/H.248: đây là giao thức điều khiển giữa MGC và MG.

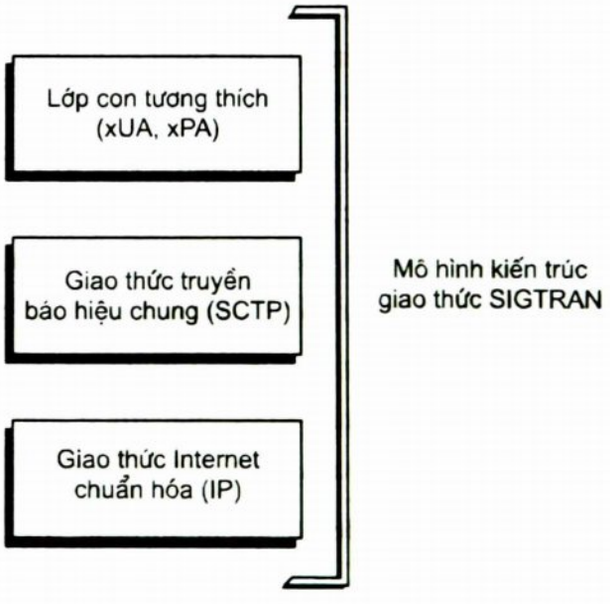
- SIP: giao thức điều khiển và khởi tạo phiên. SIP điều khiển cuộc gọi và hoạt động trong phạm vi giữa các nút MGC hoặc giữa các MGC và các máy điện thoại IP (IP Phone/Softphone).

- RTP: giao thức thời gian thực, nó tương tự như H.323 và dùng để truyền tải các gói tin phương tiện.

Rõ ràng là trong mô hình mạng tích hợp của IETF thì bộ giao thức SIGTRAN là đáng quan tâm nhất khi xem xét đến việc truyền tải lưu lượng báo hiệu SS7 qua mạng IP. Trong hình 7.1 ở trên, các đường đứt nét biểu thị các liên kết ở đó các giao thức SIGTRAN được sử dụng.

7.2 TỔNG QUAN VỀ GIAO THỨC SIGTRAN

SIGTRAN là một nhóm làm việc trực thuộc IETF, được hình thành vào năm 1999 với nhiệm vụ thiết lập một kiến trúc dùng để truyền tải các dữ liệu báo hiệu thời gian thực qua mạng IP. Nhiệm vụ chủ yếu của nó không chỉ về mặt kiến trúc mà còn bao gồm cả việc định nghĩa một bộ giao thức dùng cho việc truyền tải các bản tin báo hiệu SS7 và ISDN qua mạng IP.



Hình 7.2: Kiến trúc giao thức SIGTRAN

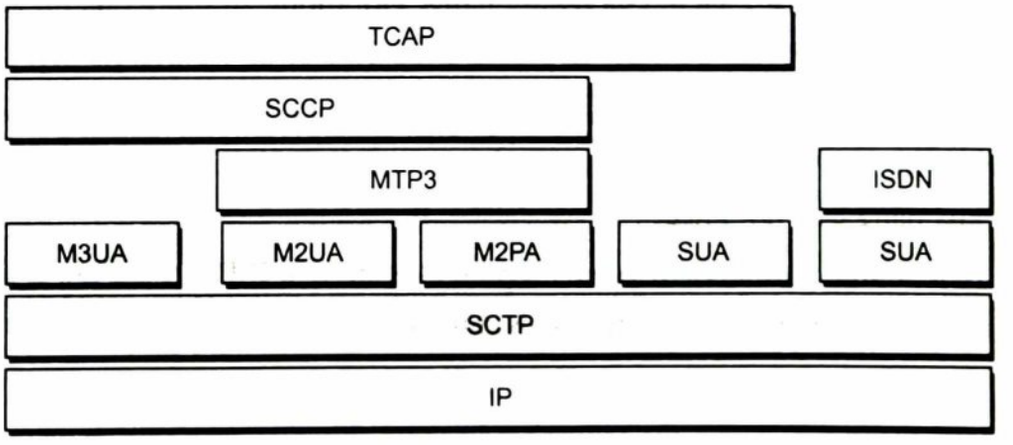
Nhóm làm việc đã đưa ra mô hình kiến trúc của giao thức SIGTRAN, gồm ba thành phần sau:

• **Giao thức Internet chuẩn hóa**

• **Giao thức truyền báo hiệu chung.** Giao thức này hỗ trợ một tập hợp chung của các chức năng truyền tải báo hiệu tin cậy. Đặc biệt trong đó phải kể đến giao thức truyền tải điều khiển luồng SCTP (Stream Control Transmission Protocol) là một giao thức truyền tải mới được định nghĩa bởi IETF.

• **Lớp con tương thích.** Các lớp tương thích có nhiệm vụ hỗ trợ các hàm nguyên thuỷ cụ thể chẳng hạn như các chỉ thị quản lý yêu cầu bởi một giao thức báo hiệu ứng dụng đặc biệt. Các giao thức lớp con tương thích mới được IETF định nghĩa gồm: M2PA (MTP2-User peer-to-peer Adaptation), M2UA (MTP2-User Adaptation), M3UA (MTP3-User Adaptation), SUA (SCCP-User Adaptation) và IUA (ISDN User Adaptation). Chú ý rằng tại một thời điểm chỉ có duy nhất một giao thức được thực hiện.

Như vậy bộ giao thức SIGTRAN được hình thành từ một lớp truyền tải mới - giao thức truyền tải điều khiển luồng (SCTP) và một tập hợp các lớp tương thích (UA - User Adaptation), các lớp tương thích này cung cấp các dịch vụ giống như các tầng thấp của mạng báo hiệu SS7 và ISDN. Hình 7.3 dưới đây minh họa chi tiết hơn về các giao thức của SIGTRAN.



Hình 7.3: Bộ giao thức SIGTRAN

Trong hình 7.3 vùng xám đậm (lớp 2 và lớp 3 từ dưới lên) chính là các giao thức mới của SIGTRAN, trong khi các vùng nhạt hơn là các giao thức đang tồn tại. Để đơn giản ISUP không được chỉ ra ở đây (thông thường ISUP chạy qua MTP3 hoặc M3UA). Các lớp thích ứng UA được đặt tên theo dịch vụ mà chúng thay thế chứ không căn cứ vào đối tượng sử dụng dịch vụ đó. Ví dụ M3UA tương thích với SCTP để cung cấp nhiều dịch vụ của lớp MTP3 chứ không phải chỉ cung cấp một dịch vụ cho MTP3. Tất cả các lớp thích ứng SIGTRAN đều phục vụ cho một số một số mục đích chung sau:

• Dùng để vận chuyển các giao thức báo hiệu lớp cao hơn thông qua cơ chế truyền tải tin cậy dựa trên nền IP.

• Cung cấp lớp dịch vụ tương tự tại giao diện của mạng PSTN tương ứng. Chẳng hạn, ít nhất thì M3UA phải khiến cho người dùng của nó nhìn nhận nó giống như MTP3 về mặt dịch vụ (M3UA không thực sự thay thế các tính năng và hoạt động của MTP3).

• Các lớp thích ứng hoàn toàn trong suốt đối với người dùng. Người sử dụng dịch vụ sẽ không nhận thấy rằng lớp thích ứng đã thay thế giao thức ban đầu (mặc dù điều này phụ thuộc chủ yếu vào việc thực hiện).

• Loại bỏ các lớp SS7 mức thấp càng nhiều càng tốt.

SIGTRAN hiện thời đưa ra sáu lớp thích ứng sau:

1. **M2UA:** cung cấp các dịch vụ của MTP2 trong mô hình client-server, chẳng hạn như SG - tới - MGC. Đối tượng sử dụng của nó sẽ là MTP3.
2. **M2PA:** cung cấp các dịch vụ của MTP2 theo mô hình ngang hàng, ví dụ như các kết nối SG - tới - SG. Đối tượng sử dụng của nó là MTP3.
3. **M3UA:** cung cấp các dịch vụ của MTP3 trong cả hai kiểu kiến trúc: client-server (SG - tới - MGC) và ngang hàng. Đối tượng sử dụng của nó sẽ là SCCP và/hoặc ISUP.
4. **SUA:** cung các các dịch vụ của SCCP trong kiến trúc ngang hàng, ví dụ SG - tới - IP SCP. Đối tượng sử dụng của nó là TCAP hoặc phần ứng dụng dựa trên khả năng trao đổi khác.
5. **IUA:** cung cấp các dịch vụ của lớp liên kết dữ liệu ISDN (LAPD). Đối tượng sử dụng của nó là một thực thể ISDN mức 3.
6. **V5UA:** cung cấp các dịch vụ của giao thức V5.2

   Có một điều đáng lưu ý là khung làm việc của bộ giao thức SIGTRAN là khá mềm dẻo, do đó cho phép chúng ta có thể thêm vào các lớp mới khi cần. Mỗi UA có một ứng dụng cụ thể và điều này sẽ được đề cập đến trong các phần sau.

7.3 GIAO THỨC SCTP

7.3.1 Vấn đề đối với giao thức TCP

Giao thức điều khiển truyền dẫn TCP (Transmission Control Protocol) thực hiện một khối lượng lớn các dịch vụ, nó đóng vai trò như là một giao thức truyền tải dữ liệu chính trong các mạng IP. TCP là giao thức theo kiểu dòng byte, nó coi dữ liệu gửi đi là

một dòng byte chứ không phải là các gói tin. Do đó nó đảm bảo dữ liệu được phân phát theo đúng thứ tự của các byte gửi nhận. Điều này khiến nó thích hợp cho việc truyền các dữ liệu cỡ lớn và không có cấu trúc chẳng hạn như một tệp tin hoặc một thư điện tử. Tuy nhiên bởi vì nó đã được định nghĩa khá lâu và được thiết kế như là một giao thức hướng gói (packet-oriented protocol) cho nên TCP có một vài hạn chế trong các ứng dụng mới gần đây. Các hạn chế đó là:

- **Do các cơ chế tin cậy:** TCP chuyển giao dữ liệu một cách tin cậy (thông qua cơ chế xác nhận) và kiểm soát chặt chẽ thứ tự của việc chuyển giao này (qua cơ chế số thứ tự). Một số ứng dụng cần truyền tải dữ liệu tin cậy mà không cần duy trì thứ tự, trong khi đó thì một số khác lại chỉ cần tuân thủ một phần. Trong cả hai trường hợp này thì sự tắc nghẽn tiêu đề dòng (head-of-line) gây ra bởi TCP đã làm tăng độ trễ không cần thiết.

- **Các vấn đề về thời gian thực:** Cơ chế xác nhận đề cập ở trên (mà làm tăng thêm trễ không mong muốn) khiến cho giao thức TCP không còn thích hợp với các ứng dụng thời gian thực nữa.

- **TCP sockets:** Sự hạn chế về phạm vi của các TCP socket đã gây khó khăn cho khả năng sẵn sàng truyền dữ liệu ở mức cao sử dụng các multi-homed host.

- **Các vấn đề về an ninh:** TCP là giao thức tương đối dễ bị xâm nhập hoặc tấn công.

Tất cả các hạn chế trên của TCP đều có liên quan đến việc cố gắng truyền báo hiệu số 7 qua mạng IP và điều này là động lực trực tiếp thúc đẩy sự phát triển của giao thức SCTP như là một giao thức truyền tải mới cho SIGTRAN. SCTP không chỉ được phát triển riêng cho SIGTRAN, do đó SCTP có thể là một giải pháp tốt đáp ứng yêu cầu của các ứng dụng khác.

7.3.2 Giao thức truyền dẫn điều khiển luồng SCTP

SCTP là một giao thức truyền tải IP mới, nó tương đương với TCP và UDP (User Datagram Protocol), hiện thời nó cung cấp các chức năng lớp truyền tải cho rất nhiều các ứng dụng dựa trên nền Internet. SCTP đã được phê chuẩn bởi IETF như là một khuyến nghị chuẩn (được đề cập cụ thể trong RFC 2960).

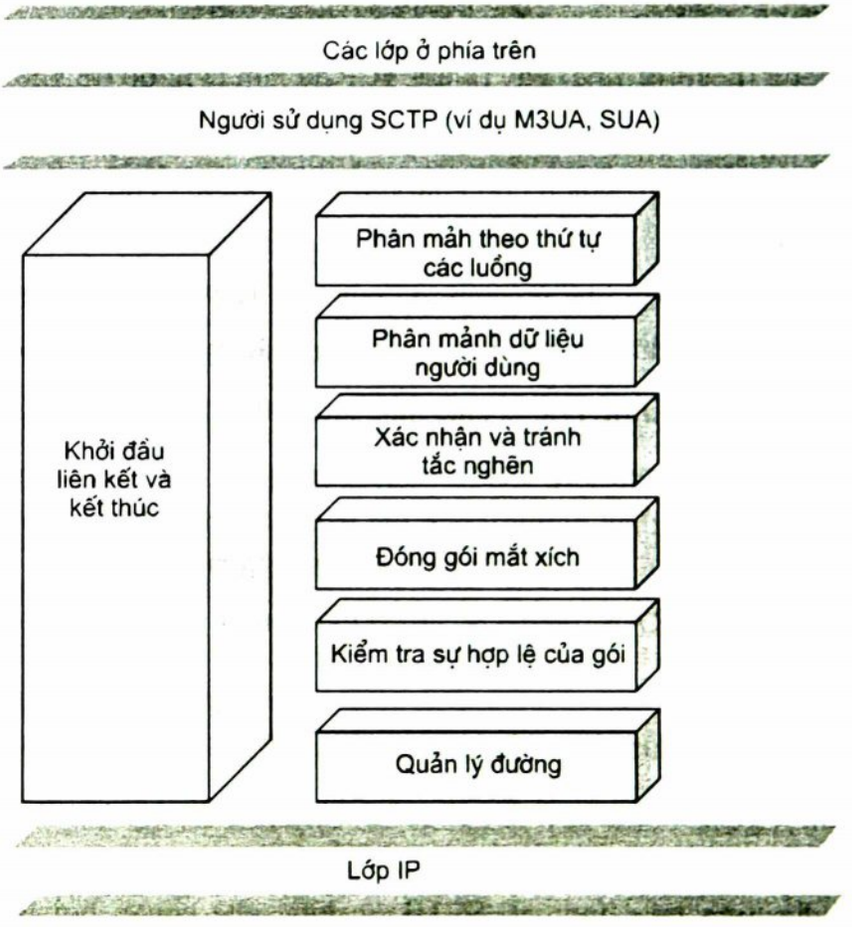
Kiến trúc của SCTP

Xét về mặt kiến trúc thì SCTP là một lớp nằm giữa lớp tương thích người dùng SCTP (hình 7.3) và một dịch vụ mạng gói phi kết nối chẳng hạn như IP. Dịch vụ cơ bản đưa ra bởi SCTP là truyền tải tin cậy các bản tin của người dùng giữa các người dùng SCTP ngang cấp. SCTP là giao thức hướng kết nối do vậy nó phải thiết lập một liên kết (association) giữa hai điểm đầu cuối trước khi truyền dữ liệu.

Chức năng của SCTP

Dịch vụ truyền tải SCTP có thể được phân chia thành nhiều chức năng như mô tả trên hình 7.4. Các chức năng cụ thể bao gồm:

• **Khởi đầu liên kết và kết thúc (Association startup and takedown)**: Một liên kết được hình thành bởi yêu cầu từ người dùng SCTP. Cơ chế cookie được áp dụng trong suốt quá trình khởi đầu để chống lại các cuộc tấn công. Cơ chế cookie sử dụng thủ tục bắt tay gồm bốn bước, hai bước cuối được phép mang dữ liệu của người dùng dành cho việc thiết lập nhanh.



Hình 7.4: Các chức năng của SCTP

SCTP cho phép kết thúc một liên kết đang hoạt động theo cách thông thường (ví dụ bằng lệnh Shutdown) dựa trên yêu cầu của người dùng SCTP. Nó cũng đưa ra phương pháp ngắt kết nối theo kiểu bất thường (ví dụ bằng lệnh hủy bỏ Abort) dựa trên yêu cầu của người dùng hoặc căn cứ vào tình trạng của lỗi phát hiện thấy bên trong lớp SCTP.

SCTP không hỗ trợ trạng thái nửa mở (half-open) như giao thức TCP, trạng thái mà một bên có thể tiếp tục gửi dữ liệu trong khi phía kia đã đóng. Khi cả hai phía đầu

cuối đều ngắt thì liên kết ở mỗi điểm sẽ ngừng tiếp nhận dữ liệu mới từ phía người dùng của nó và chỉ phân phát dữ liệu còn lại ở hàng đợi tại thời điểm ngắt.

• **Phân phối theo thứ tự trong các luồng (Sequenced Delivery within Streams)**: Thuật ngữ "luồng" (Stream) dùng trong SCTP ám chỉ một chuỗi có thứ tự các bản tin của người dùng được chuyển đến giao thức lớp cao hơn, có sự phân biệt so với các bản tin khác trong cùng một luồng. Điều này trái ngược hoàn toàn so với khái niệm luồng trong giao thức TCP, ở đây đề cập đến một chuỗi có thứ tự của các byte.

Người dùng SCTP có thể xác định rõ tại thời điểm khởi đầu liên kết số lượng luồng dữ liệu được hỗ trợ bởi liên kết. Con số này được thoả thuận với đầu cuối phía xa. Các bản tin của người dùng được gắn với các số luồng. Bên trong SCTP gán một số thứ tự luồng cho mỗi bản tin truyền đến nó bởi người dùng SCTP. Ở phía đầu thu SCTP đảm bảo các bản tin được chuyển đến người dùng SCTP theo đúng thứ tự trong luồng đã gán. Tuy nhiên nếu có một luồng bị khóa (block) và đang chờ bản tin người dùng kế tiếp theo thứ tự thì việc phân phát các luồng khác vẫn có thể được tiến hành bình thường. SCTP cung cấp cơ chế bỏ qua thứ tự phân phát dịch vụ. Các bản tin của người dùng gửi đi sử dụng cơ chế này sẽ được chuyển đến người dùng SCTP miễn là chúng được nhận.

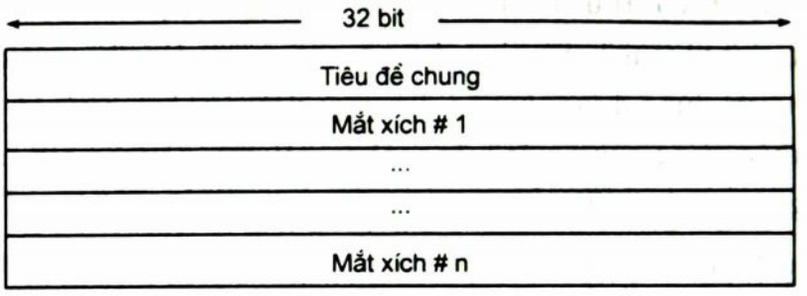
• **Phân mảnh dữ liệu người dùng (User Data Fragmentation)**: SCTP hỗ trợ sự phân mảnh bản tin của người dùng nhằm đảm bảo rằng gói tin SCTP được truyền đến lớp dưới theo tuyến MTU (multiple-tenant unit). Tại phía đầu thu các mảnh dữ liệu này lại được tái hợp lại thành bản tin hoàn chỉnh trước khi được chuyển đến người dùng SCTP.

• **Xác nhận và tránh tắc nghẽn (Acknowledgement and Congestion Avoidance)**: SCTP gán một số thứ tự truyền (TSN - transmission sequence number) cho mỗi bản tin đã phân mảnh hoặc không phân mảnh của người dùng. Phía đầu cuối thu xác nhận tất cả các số thứ tự TSN đã nhận được, kể cả khi có các khoảng trống trong chuỗi. Với cách này thì việc chuyển giao tin cậy được tách biệt hoàn toàn về mặt chức năng so với việc chuyển giao theo thứ tự luồng.

Chức năng xác nhận và tránh tắc nghẽn có trách nhiệm phải truyền lại các gói tin khi có xác nhận là gói tin vẫn chưa đến đích. Việc truyền lại gói được quyết định bởi các thủ tục tránh tắc nghẽn, các thủ tục này tương tự như ở giao thức TCP.

• **Đóng gói mắt xích (Chunk Bundling)**: Gói tin SCTP được chuyển đến lớp dưới bao gồm một tiêu đề chung và nối tiếp là một hoặc rất nhiều mắt xích (chunk). Mỗi mắt xích có thể chứa hoặc dữ liệu của người dùng hoặc thông tin điều khiển SCTP. Người dùng SCTP có quyền lựa chọn để đóng gói nhiều hơn một bản tin người dùng vào trong một gói SCTP đơn. Chức năng đóng gói mắt xích của SCTP có nhiệm vụ hợp

nhất các gói tin SCTP hoàn chỉnh và tách nó tại phía đầu thu. Hình 7.5 dưới đây miêu tả cấu trúc chung của một gói SCTP:



Hình 7.5: Cấu trúc của một gói tin SCTP

• **Kiểm tra sự hợp lệ của gói (Packet Validation)**: Trong phần tiêu đề chung của SCTP có 2 trường đó là trường thẻ xác minh (Verification Tag) bắt buộc và một trường tổng kiểm tra (checksum - 32 bit). Giá trị của thẻ xác minh được chọn bởi các đầu cuối của liên kết trong quá trình khởi đầu liên kết. Các gói nhận được mà không có giá trị của thẻ này thì sẽ bị đào thải ngay lập tức nhằm hạn chế các cuộc tấn công giả mạo đồng thời loại bỏ các gói tin SCTP cũ của liên kết trước đó còn sót lại. Thêm vào đó, thuật toán tổng kiểm tra checksum Adler-32 thiết lập bởi người gửi gói tin SCTP cung cấp sự bảo vệ để chống lại sự gián đoạn dữ liệu của mạng. Phía người nhận cũng bằng thuật toán checksum Adler-32 nếu kiểm tra thấy gói tin SCTP không hợp lệ thì nó sẽ tự động loại bỏ gói tin này.

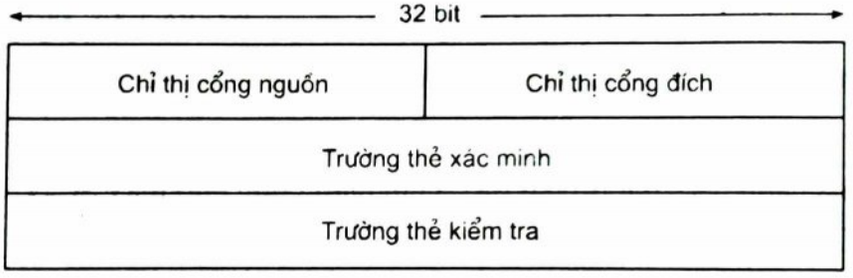
• **Quản lý đường (Path Management)**: Chức năng quản lý đường của SCTP sẽ chọn ra địa chỉ đích cần truyền đối với mỗi gói SCTP tại đầu ra dựa trên các lệnh của người dùng SCTP và tình trạng hiện thời của đích. Tại thời điểm khởi đầu liên kết, một đường dẫn chính được thiết lập cho mỗi điểm đầu cuối SCTP và nó được sử dụng để gửi các gói SCTP. Ở phía đầu thu, chức năng quản lý đường có trách nhiệm xác minh sự tồn tại của liên kết SCTP hợp lệ đối với gói tin SCTP đến trước khi tiếp tục xử lý nó.

Chú ý: Chức năng quản lý đường và kiểm tra sự hợp lệ của gói được tiến hành đồng thời.

Khuôn dạng tiêu đề chung của SCTP

Hình 7.6 dưới đây mô tả khuôn dạng tiêu đề chung của SCTP

• **Trường chỉ thị cổng nguồn/đích (Source/Destination Port Number)** (16 bit): Chỉ thị số cổng của người gửi/nhận gói SCTP.



Hình 7.6: Khuôn dạng tiêu đề chung của SCTP

• **Trường thẻ xác minh (Verification Tag)** (32 bit): Người nhận gói tin này sử dụng thẻ xác minh để xác nhận tính hợp lệ của người gửi gói tin SCTP.

• **Trường tổng kiểm tra (Checksum)** (32 bit): Trường này chứa tổng kiểm tra của gói SCTP này. SCTP sử dụng thuật toán Adler-32 để tính tổng kiểm tra.

SCTP có các đặc điểm chính sau:

• Nó là giao thức Unicast (điểm - điểm). Dữ liệu được trao đổi giữa hai điểm kết cuối.

• SCTP định nghĩa các bộ định thời có khoảng thời gian ngắn hơn nhiều so với các định thời của TCP.

• Đảm bảo việc truyền tải tin cậy dữ liệu của người dùng, phát hiện kịp thời khi mà dữ liệu bị gián đoạn hay không đúng thứ tự và tiến hành sửa chữa nếu cần thiết.

• Thích ứng với tốc độ truyền, phản ứng kịp thời đối với tắc nghẽn mạng và căn cứ vào đó để điều chỉnh việc truyền lại.

• Hỗ trợ multi-homing. Mỗi điểm kết cuối SCTP có thể được biết bởi rất nhiều địa chỉ IP. Việc định tuyến đến một địa chỉ là hoàn toàn độc lập đối với các địa chỉ khác và nếu một tuyến nào đó không sẵn sàng thì tuyến khác sẽ được sử dụng.

• SCTP sử dụng thủ tục khởi đầu dựa trên các cookie, để ngăn chặn các cuộc tấn công từ chối dịch vụ.

• Hỗ trợ đóng gói dữ liệu. Các bản tin SCTP riêng lẻ có thể chứa rất nhiều mắt xích của dữ liệu, mỗi mắt xích này lại có thể chứa toàn bộ một bản tin báo hiệu.

• Hỗ trợ sự phân mảnh dữ liệu, nghĩa là một bản tin báo hiệu có thể được chia thành nhiều bản tin SCTP để phù hợp với đơn vị dữ liệu giao thức PDU (Protocol Datagram Unit).

• Nó là giao thức hướng bản tin (message-oriented), nó định nghĩa cấu trúc khung của dữ liệu. Trái lại TCP không quan tâm tới cấu trúc của dòng byte mà nó truyền.

• SCTP có tính năng đa luồng (multi-streaming). Dữ liệu được phân chia thành nhiều luồng, mỗi luồng lại được phân phát độc lập với nhau về thứ tự. TCP không có đặc điểm này.

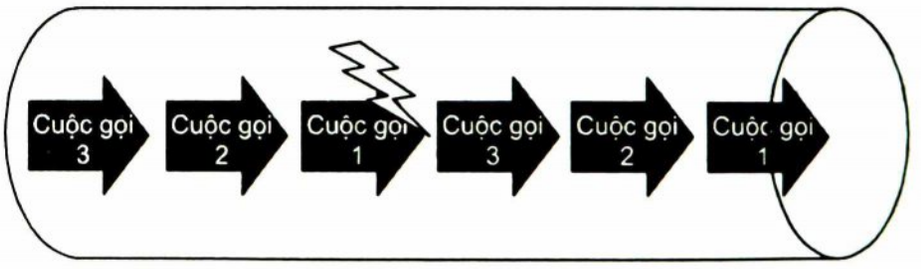
Xét về khía cạnh bảo mật thì hai đặc điểm cuối cùng cho thấy SCTP thích hợp hơn nhiều so với TCP để mang các bản tin báo hiệu.

Việc truyền các khung dữ liệu đã được cấu trúc và đóng gói là một việc khá dễ dàng đối với các lớp thích ứng UA (cũng như bất kì người dùng SCTP tiềm năng nào khác). Ở đây giao thức truyền tải đã làm tất cả mọi việc để phân chia luồng dữ liệu vào trong các đoạn (segment) bản tin, làm giảm nhẹ gánh nặng cho người dùng trong việc diễn giải luồng byte tiếp theo.

Bên cạnh đó nhờ định nghĩa cấu trúc của bản tin nên quá trình đóng gói và phân mảnh dữ liệu diễn ra được thuận lợi hơn.

Tuy nhiên tính năng đa luồng (multi-streaming) mới là yếu tố hấp dẫn chính của SCTP (thậm chí tên của giao thức đã ám chỉ điều này: Giao thức truyền dẫn điều khiển luồng). Đặc điểm này rõ ràng được thiết kế để cho phép các User phân chia một kết nối IP đơn giữa hai điểm kết cuối thành các luồng dữ liệu lô-gíc riêng biệt và gán cho mỗi luồng đó một ứng dụng hoặc một tài nguyên cụ thể. Điều này đảm bảo rằng các lỗi và các trễ trên một luồng sẽ không ảnh hưởng đến sự phân phát bình thường của các luồng khác.

Chúng ta hãy xem xét ví dụ của giao thức ISUP. ISUP mang các bản tin báo hiệu cho rất nhiều nguồn tài nguyên của PSTN (chẳng hạn như các mạch trung kế, ở đây chúng ta coi mỗi mạch trung kế như là một cuộc gọi điện thoại). Bây giờ chúng ta xem xét trường hợp có 3 cuộc gọi đang trong tiến trình thực hiện, sau đó một trong số chúng bị mất dữ liệu thì liệu điều này gây ra trễ trong việc truyền các bản tin khác liên quan đến cuộc gọi hay không?



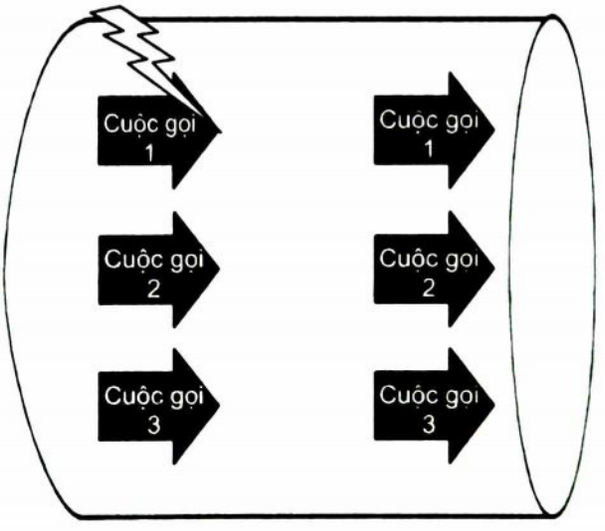
Hình 7.7: ISUP qua TCP

Hình 7.7 chỉ ra tình huống này, ở đây ISUP được truyền thông qua TCP. Một đường ống dữ liệu đơn mang tất cả các bản tin ISUP của cả ba cuộc gọi.

Giả sử các bản tin thiết lập cho ba cuộc gọi được xác nhận đúng. Sau đó tất cả các cuộc gọi đều ở trong tiến trình thực hiện. Tiếp đến là quá trình giải phóng cuộc gọi, quá trình này phải được tiến hành theo đúng thứ tự. Nhưng không may là bản tin giải phóng của cuộc gọi 1 lại bị mất. Hậu quả là các bản tin giải phóng của hai cuộc gọi (2 và 3) còn lại sẽ bị trễ do TCP mất thời gian cho việc ghi lại sự tổn thất và tìm cách khôi phục dữ liệu. Chính điều này đã gây ra trễ rất lớn đối với việc giải phóng cuộc gọi 2 và 3, điều không thể chấp nhận được đối với mạng PSTN.

Bây giờ chúng ta sẽ xem xét tiếp trường hợp ISUP được truyền qua SCTP. Hình 7.8 minh họa trường hợp này.

Trong trường hợp này, sự tổn thất của bản tin liên quan đến cuộc gọi 1 chỉ ảnh hưởng duy nhất đến luồng dữ liệu đó. Cuộc gọi 2 và 3 vẫn hoạt động như bình thường. Các nguồn tài nguyên liên quan đến cuộc gọi 1 vẫn sẽ được liên hệ cho đến khi giao thức truyền tải khôi phục dữ liệu đã mất. Tình huống này không khác mấy so với trường hợp của các giao thức truyền tải trong mạng PSTN truyền thống.



Hình 7.8: ISUP qua SCTP

7.3.3 M2UA

Giao thức thích ứng người dùng phần chuyển giao bản tin mức 2 (M2UA -MTP2 user adaptation) được sử dụng để chuyển giao các bản tin báo hiệu SS7 phần người dùng MTP2 (ví dụ các bản tin MTP3) qua mạng IP sử dụng các dịch vụ của SCTP. Cụ thể M2UA truyền dữ liệu của người dùng MTP2 giữa một lớp MTP2 đặt tại SG và một

lớp MTP3 đặt tại MGC. Như vậy nó hoạt động giống như mô hình máy trạm/máy chủ trong đó MGC là máy trạm và SG là máy chủ. M2UA hỗ trợ:

- **Ranh giới giao diện giữa MTP2/MTP3**

  Giao diện SS7 giữa MTP3/MTP2 (MTP2-User) vẫn được giữ lại tại điểm đầu cuối trong mạng IP, do đó lớp giao thức M2UA được yêu cầu cung cấp cho các User của nó tập các dịch vụ tương đương với các dịch vụ mà MTP2 đã cung cấp cho MTP3.

  Chú ý: thuật ngữ giao diện ở đây được dùng với nghĩa là một liên kết (hay kênh) báo hiệu SS7.

- **Giao tiếp giữa các mô-đun quản lý lớp đặt tại SG và MGC**

  M2UA cung cấp một số bản tin nhằm hỗ trợ cho giao tiếp giữa các mô-đun quản lý lớp đặt tại SG và MGC diễn ra thuận lợi hơn. Để thuận tiện cho việc báo cáo lỗi phát sinh bởi quá trình "*backhauling*" phần trao đổi bản tin mức 3 (MTP3), bản tin sau đây được định nghĩa:

  M-ERROR: bản tin này được sử dụng để chỉ thị lỗi của bản tin mà M2UA nhận được.

  Backhauling: là quá trình truyền tải báo hiệu từ điểm giao diện của luồng dữ liệu liên kết (ví dụ cổng chức năng SG ở MGU) trở về điểm xử lý cuộc gọi (ví dụ MGCU), nếu đây không phải là cuộc gọi nội hạt.

- **Hỗ trợ quản lý các liên kết đang hoạt động giữa SG và MGC**

  Lớp M2UA ở SG có nhiệm vụ giữ trạng thái của các ASP (Application Server Process - Tiến trình máy chủ ứng dụng) đã được cấu hình. Một tập các hàm nguyên thủy (primitives) giữa M2UA và mô-đun quản lý lớp (Layer Management) được định nghĩa dưới đây nhằm giúp cho mô-đung này quản lý liên kết giữa SG và MGC. Lớp M2UA có thể dựa vào chỉ dẫn của mô-đun quản lý lớp để thiết lập một liên kết SCTP với một nút M2UA ngang cấp. Thủ tục này có thể được thực hiện thông qua hàm M-SCTP ESTABLISH.

- M-SCTP ESTABLISH: hàm này được sử dụng để yêu cầu, chỉ thị và xác nhận việc thiết lập của một liên kết SCTP với một nút M2UA ngang cấp.

- M-SCTP_RELEASE: các hàm này được sử dụng để yêu cầu, chỉ thị và xác nhận việc giải phóng của một liên kết SCTP với một nút M2UA ngang cấp.

Bên cạnh đó thì M2UA cũng có thể cần phải thông báo về tình trạng của liên kết SCTP cho mô-đun quản lý lớp. Việc này được thực hiện bởi hàm nguyên thủy dưới đây:

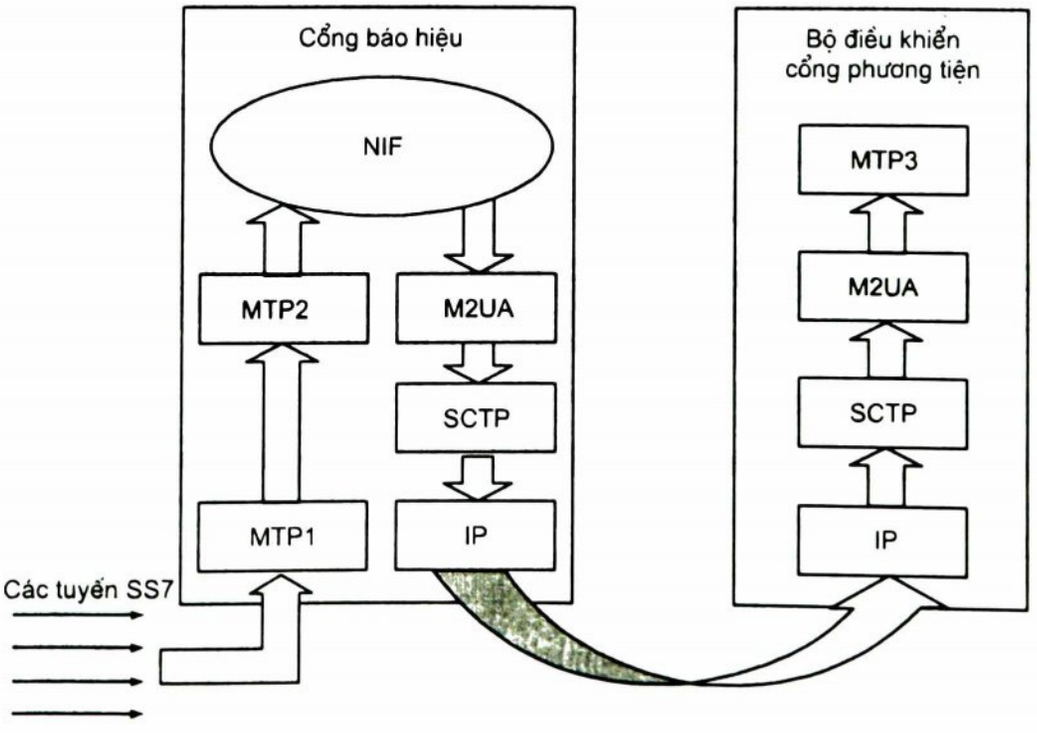
- M-SCTP_STATUS: yêu cầu và chỉ thị về tình trạng của liên kết SCTP nằm dưới.

Ngoài ra mô-đun quản lý lớp cũng có thể cần thông báo cho lớp M2UA về tình trạng của AS (Application Server - Máy chủ dịch vụ)/ASP (ví dụ như đang hoạt động hay bị lỗi v.v...), nhờ đó các bản tin có thể được trao đổi giữa các lớp M2UA đồng cấp để ngừng lưu lượng đến người dùng M2UA nội hạt. Điều này được thực hiện thông qua hàm M-ASP_STATUS.

- M-ASP_STATUS: trạng thái của ASP được lưu trữ bên trong lớp M2UA ở cả hai phía SG và MGC. M-ASP_STATUS được sử dụng bởi mô-đun quản lý lớp để yêu cầu báo cáo về trạng thái của ASP từ lớp M2UA. Hàm này cũng có thể được sử dụng để chỉ thị trạng thái của ASP.

- M-ASP_MODIFY: mô-đun quản lý lớp sử dụng hàm này để thay đổi trạng thái của ASP. Nói cách khác mô-đun quản lý lớp ở phía ASP sử dụng hàm này để khởi đầu các thủ tục ASPM (ASP Modify).

- M-AS_STATUS: được sử dụng bởi mô-đun quản lý lớp để yêu cầu trạng thái của AS. Nó cũng có thể được dùng để chỉ thị trạng thái của AS.



Hình 7.9: M2UA

Mô hình kiến trúc trong đó sử dụng M2UA được chỉ ra trên hình 7.9. Trên hình 7.9 mô tả các chức năng và quan hệ giữa MGC và SG. Tại cổng báo hiệu SG khối chức năng NIF (Nodal Interworking Function) là chức năng liên kết nút.

Xét về mặt hiệu quả thì MTP3 ở MGC chính là User của MTP2 ở SG. Cả MTP2 và MTP3 đều không nhận thấy rằng chúng được đặt cách xa nhau. Với quá trình xử lý này thì các bản tin báo hiệu được truyền qua mạng IP từ đỉnh của một lớp SS7 đến đáy của một lớp khác được miêu tả như là quá trình "*backhauling*".

Còn User của MTP3 tại MGC thông thường sẽ là ISUP. Kiến trúc này có thể áp dụng trong hầu hết các trường hợp sau:

• Mật độ của các kênh báo hiệu SS7 tại một điểm vật lý cụ thể trong mạng thấp.

• Có một số lượng lớn các chức năng của SG tách biệt về mặt vật lý (vì rằng các liên kết SS7 về mặt vật lý đặt ở cách xa nhau).

• Chức năng của SG được đặt cùng (co-located) với MG (thường căn cứ vào một hoặc nhiều điều kiện trước đó).

Cấu hình cụ thể của SIGTRAN chắc chắn là khá phổ biến ở các mạng châu Âu, nơi mà các kênh báo hiệu cùng chia sẻ các phương tiện vật lý với các mạch thoại.

Ở phía SG, việc miêu tả M2UA tại cùng mức với MTP2 dễ gây ra sự hiểu lầm nhỏ. Theo nhiều cách khác nhau thì M2UA chính là người dùng của MTP2. Như vậy M2UA phải có trách nhiệm đối với các hoạt động khởi đầu của giao thức mà thông thường được xử lý bởi MTP3, chẳng hạn như:

+ Hoạt hóa và không hoạt hóa liên kết

+ Các yêu cầu về số thứ tự

+ Các thủ tục cập nhật bộ đệm truyền/truyền lại của MTP2

+ Làm sạch bộ đệm

Chức năng này có thể được đặt tại NIF (Nodal Interworking Function).

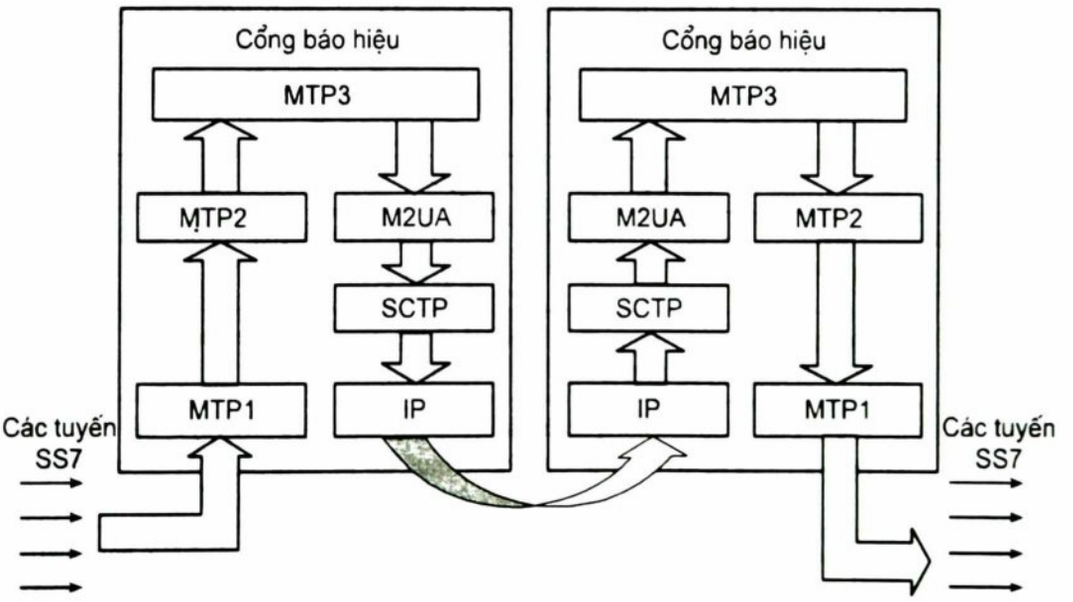
7.3.4 M2PA

Giao thức thích ứng ngang cấp người dùng của phần chuyển giao bản tin mức 2 (M2PA - MTP2 User peer-to-peer Adaptation) sử dụng các dịch vụ của SCTP để hỗ trợ việc truyền các bản tin báo hiệu MTP3 của hệ thống SS7 qua mạng IP. M2PA có khả năng xử lý đầy đủ bản tin MTP3 và quản lý mạng giữa bất kì hai nút SS7 nào giao tiếp với nhau qua mạng IP. M2PA hỗ trợ:

- Điều khiển liên tục hoạt động của các giao thức MTP3 ngang cấp qua một kết nối với mạng IP.
- Ranh giới giao diện giữa MTP2 và MTP3, quản lý các liên kết truyền tải SCTP và lưu lượng thay cho các liên kết của MTP2.
- Báo cáo kịp thời các thay đổi về trạng thái đến phần quản lý.

M2PA là giao thức ngang cấp tương đương của M2UA. Nó cung cấp một đường liên kết giữa MTP2 và MTP3 đặt xa nhau, nó thay thế đường liên kết mức thấp giữa MTP2 và MTP3. Người dùng của M2PA là MTP3 tại cả hai đầu cuối của kết nối (đối với M2UA thì một user là MTP3 và User kia là SG IWF).

M2PA là phương tiện giúp cho các lớp MTP3 cùng cấp ở các SG có thể giao tiếp trực tiếp với nhau. Thực chất nó là sự mở rộng của hệ thống báo hiệu SS7 thông qua mạng IP. Mô hình kiến trúc sử dụng M2PA được chỉ ra trên hình 7.10.



Hình 7.10: M2PA

Kiến trúc này áp dụng cho kết nối từ SG tới SG, nó đóng vai trò như chiếc cầu nối giữa hai mạng SS7. Trong trường hợp này mỗi SG có thể kết nối với nhiều SG khác và chúng không cần biết về lớp bên trên mà chúng đang hỗ trợ. MTP3 có mặt tại mỗi SG để tham gia vào việc định tuyến và quản lý các liên kết của MTP2/M2PA. Do có sự hiện diện của MTP3 nên mỗi SG sẽ cần phải có mã điểm của riêng nó.

Việc thay thế các liên kết của MTP2 với M2UA nhằm để phân biệt với trường hợp truy cập IP SCP từ SG (dịch vụ này được cung cấp bởi SUA). Ở trường hợp của SUA, nó

được biết lớp trên của nó là TCAP (hoặc các phần ứng dụng khác), trong khi đó thì M2PA hoàn toàn không biết gì về các lớp SS7 bên trên của nó.

Sự khác nhau quan trọng về mặt chức năng của M2PA so với M2UA là M2PA bản thân nó thực sự cung cấp dịch vụ giống như MTP2. Còn M2UA chỉ cung cấp một giao diện cho dịch vụ của MTP2 ở phía đầu xa. Do đó nhiệm vụ chủ yếu của M2PA là:

- Hoạt hóa/ngừng hoạt hoá liên kết (để đáp lại các yêu cầu từ MTP3)
- Duy trì thông tin về tình trạng của liên kết
- Duy trì các số thứ tự và các bộ đệm truyền lại nhằm giúp cho sự phục hồi lại của bản tin MTP3
- Duy trì tình trạng của bộ xử lý nội hạt và từ xa

7.3.5 So sánh M2UA và M2PA

Bảng 7.1 sau đây liệt kê các thuộc tính cơ bản để so sánh M2UA và M2PA.

Bảng 7.1: So sánh M2UA và M2PA

	M2PA	M2UA
Các bản tin dữ liệu MTP3	Truyền tải các bản tin dữ liệu MTP3	Truyền tải các bản tin dữ liệu MTP3
Giao diện với MTP3	Cung cấp một giao diện giữa MTP2 bậc cao và MTP3	Cung cấp một giao diện giữa MTP2 bậc cao và MTP3
Các hàm nguyên thủy (Primitives)	Điểm báo hiệu IP xử lý các hàm nguyên thủy từ MTP3 - tới - MTP2	Điểm báo hiệu IP truyền tải các hàm nguyên thuỷ từ MTP3 - tới - MTP2 của SG (thông qua chức năng liên kết) để xử lý
Loại liên kết	Kết nối giữa SG và điểm báo hiệu IP là một liên kết SS7 (xét theo khía cạnh MTP3)	Kết nối giữa SG và điểm báo hiệu IP không phải là một liên kết SS7. Nó là sự mở rộng của MTP2 với một nút ở xa
Các mã điểm	SG là một nút SS7 với một mã điểm riêng	SG không phải là một nút SS7 và không có mã điểm
Quản lý	Các thủ tục quản lý tin cậy dựa trên MTP3	Sử dụng các thủ tục quản lý của M2UA

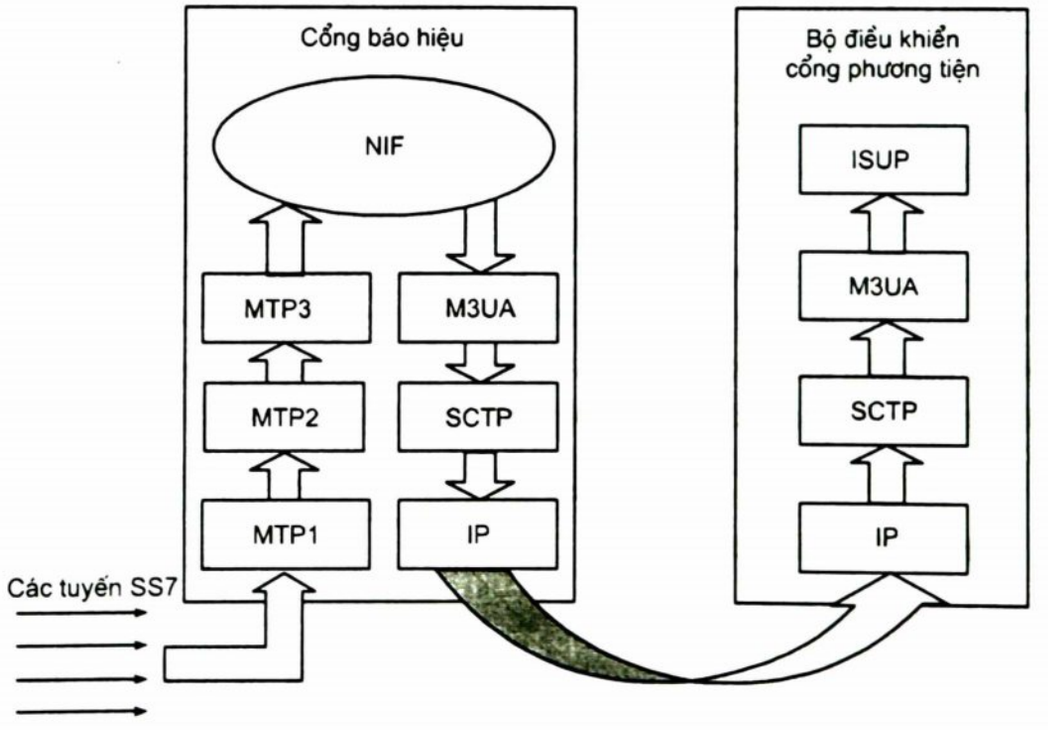
7.3.6 M3UA

Lớp thích ứng người dùng phần chuyển giao bản tin mức 3 (M3UA - MTP3 User Adaptation) định nghĩa một giao thức hỗ trợ cho việc truyền tải các bản tin báo hiệu của người dùng MTP3 (ví dụ các bản tin ISUP/SCCP...) qua mạng IP sử dụng các dịch

vụ của SCTP. Giao thức này sẽ được sử dụng ở giữa một SG và một MGC hoặc một cơ sở dữ liệu thường trú IP (IP-resident database). M3UA thích hợp cho việc truyền các bản tin của bất cứ phần người dùng MTP3 nào. Danh sách các lớp giao thức (không hạn chế) này bao gồm ISUP, SCCP, và phần người dùng điện thoại TUP. Chú ý rằng các bản tin của giao thức ứng dụng các khả năng trao đổi (TCAP) và giao thức ứng dụng mạng truy cập vô tuyến (RANAP - Radio Access Network Application Protocol) được truyền tải trong suốt bởi M3UA giống như tải trọng của SCCP bởi vì chúng là các giao thức của người dùng SCCP.

Thực chất M3UA tương tự như M2UA và nó hoạt động theo mô hình máy trạm - máy chủ nhằm cung cấp cho lớp bậc cao của hệ thống báo hiệu SS7 một giao thức để truy cập từ xa đến các lớp thấp hơn.

Nhờ có M3UA mà dịch vụ của MTP3 có thể được cung cấp tại MGC (chẳng hạn như việc hủy kết nối ISUP ở MGC). Đây cũng có thể coi là sự mở rộng của hệ thống SS7 trong mạng IP. Hình 7.11 dưới đây minh họa kiến trúc mà ở đó M3UA được sử dụng.



Hình 7.11: M3UA

MTP3 ở SG không nhận thấy rằng User của ISUP được đặt cách xa nó (NIF sẽ đăng ký bản thân nó như là ISUP với MTP3). Tương tự, ISUP ở MGC cũng sẽ không biết là nó không được phục vụ bởi một MTP3 nội hạt. Đây lại là một ví dụ nữa của quá

trình "*backhauling*" các bản tin báo hiệu của MGC từ SG. Kiến trúc này thích hợp cho các trường hợp sau:

• Mật độ của các liên kết SS7 đủ lớn để khiến cho một cổng SG đứng độc lập có thể thông qua.

• Các liên kết SS7 có thể truy cập vật lý tại một điểm đơn nào đó.

Chúng ta có thể thấy các điều kiện trên là khá phổ biến ở các mạng Bắc Mĩ nơi mà các đường SS7 tách biệt về mặt vật lý so với các mạch thoại. Ở trường hợp này, một số lượng lớn các kênh báo hiệu sẽ được tập trung trong một phương tiện vật lý đơn (ví dụ như một đường T1 chẳng hạn). Ở đây mỗi SG đều có một lớp MTP3 nội hạt và do đó nó cũng phải có mã điểm SS7 của mình.

Lớp M3UA có nhiệm vụ duy trì giao diện giữa MTP3 - ISUP qua kết nối SCTP. Các lệnh và yêu cầu truyền dữ liệu xuống tầng bên dưới của ISUP ở MGC được mang bởi M3UA và được đưa đến giao diện cao hơn của MTP3 ở SG. Còn các chỉ thị và các bản tin dữ liệu đến được chuyển lên phía trên từ MTP3 ở SG và được mang bởi M3UA (qua SCTP) đến giao diện thấp hơn của ISUP ở MGC.

7.3.7 SUA

Giao thức thích ứng người dùng SCCP (SUA - SCCP user adaptation) được dùng để truyền tải bất kỳ bản tin báo hiệu của người dùng SCCP (ví dụ TCAP, RANAP v.v...) qua mạng IP sử dụng các dịch vụ của SCTP. Giao thức này được thiết kế theo kiểu mô-đun và có tính đối xứng do đó cho phép nó hoạt động trong các kiến trúc đa dạng chẳng hạn như kiến trúc SG - tới - IP báo hiệu điểm cuối cũng như kiến trúc điểm kết cuối báo hiệu IP ngang cấp (peer-to-peer). SUA hỗ trợ:

• Trao đổi các bản tin của phần người dùng SCCP (các bản tin TCAP, RANAP...)

• Dịch vụ SCCP phi kết nối

• Dịch vụ SCCP hướng kết nối

• Quản lý các liên kết truyền tải SCTP giữa một nút SG và một hoặc nhiều nút báo hiệu dựa trên nền IP

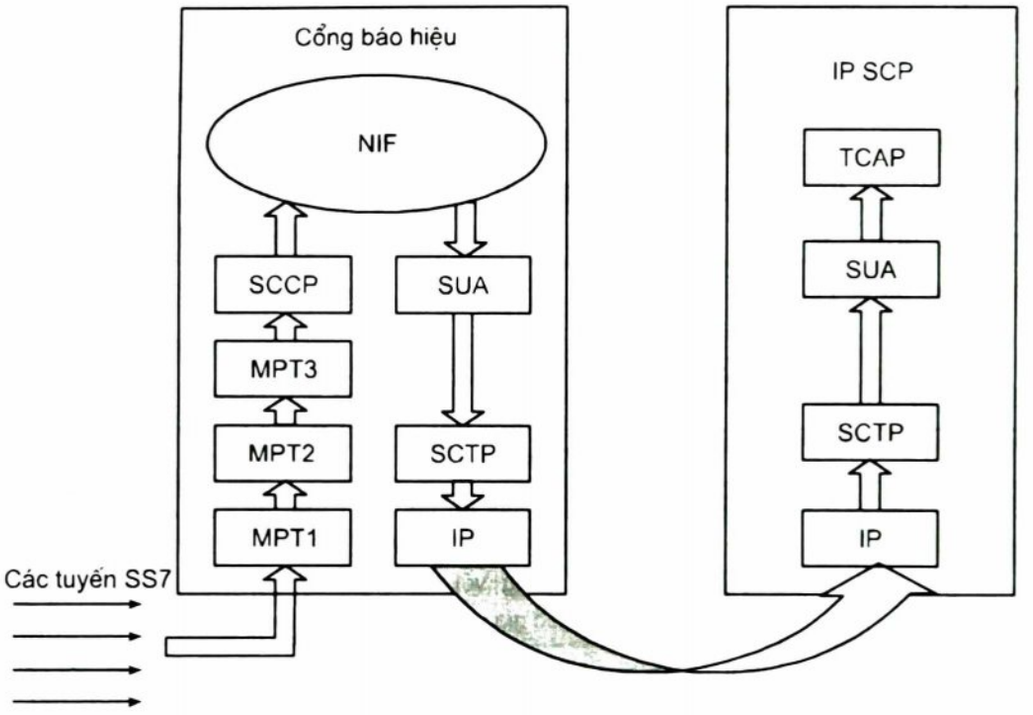
• Phân phối các nút báo hiệu IP

• Báo cáo kịp thời các thay đổi về trạng thái cho phần quản lý

SUA cung cấp phương tiện nhờ đó phần ứng dụng (chẳng hạn như TCAP) ở một nút điều khiển dịch vụ IP (IP SCP - IP Service Control Point) có thể được thực hiện

thông qua SG. Kiến trúc mạng gắn với SUA cho phép có hàng loạt các nút điều khiển dịch vụ IP SCP thông qua một SG đơn. Các IP SCP này không có lớp MTP3 nội hạt và do đó chúng không cần mã điểm SS7.

Hình 7.12 trình bày mô hình kiến trúc ở đó SUA được sử dụng giữa SG và IP SCP.



Hình 7.12: SUA

Chức năng của SUA có thể được cung cấp bởi MTP2 hoặc các lớp thích ứng MTP2. Tuy nhiên SUA có một chức năng quan trọng đó là chuyển đổi giữa địa chỉ SCCP và địa chỉ IP (tại SG). Nếu không có chức năng này thì SCCP sẽ phải có mặt ở tất cả các điểm IP SCP và mạng SS7 mở rộng sẽ cần đến thông tin của các SCCP này. SUA có thể biết được sự hiện diện của từng điểm IP SCP bằng cách cung cấp một địa chỉ SCCP để bao phủ tất cả các nút.

Dịch vụ của các cơ sở dữ liệu cá nhân được đánh địa chỉ thông qua số phân hệ SSN (Subsystem Number). SUA cung cấp dịch vụ không giống như dịch vụ biên dịch tiêu đề toàn cầu (GTT - Global Title Translation) để sắp xếp các SSN vào trong kết nối SCCP (được dùng để định tuyến các bản tin của phần ứng dụng đến điểm điều khiển dịch vụ IP SCP thích hợp).

SUA cũng là một giao thức khá mềm dẻo, nó hỗ trợ các phần ứng dụng đang chạy giữa hai nút trong toàn bộ mạng IP. Điều này đặc biệt thích hợp đối với các mạng mới ra đời. Đối với các mạng này thì một mạng báo hiệu SS7 truyền thống chạy ở phía dưới

có thể không cần thiết. Trong trường hợp này ngăn xếp IP SCP sẽ giống nhau ở cả hai nút (dựa trên IP).

Ngoài ra SUA còn cho phép truy cập đến các cơ sở dữ liệu dịch vụ trong mạng SS7 từ mạng IP. Khi đó mô hình kiến trúc sẽ tương tự như hình 7.12.

7.3.8 SO SÁNH M3UA VÀ SUA

Nói chung ngăn xếp giao thức dựa trên SUA không phức tạp và hiệu quả hơn nhiều so với ngăn xếp giao thức dựa trên SCCP và M3UA. Bởi vậy, SUA có thể tăng cường hiệu quả của mạng lõi và cung cấp các phương tiện cho giúp cho việc thực hiện được đơn giản hơn. Để thấy rõ hơn những nhận xét trên, bảng 7.2 liệt kê một số thuộc tính cơ bản của M3UA và SUA.

Bảng 7.2: So sánh M3UA và SUA

	M3UA	**SUA**
Các đặc điểm của SCCP (SCCP Flavors)	Điểm báo hiệu cần phải hỗ trợ các đặc điểm khác nhau của SCCP nếu nó liên kết hoạt động với các hệ thống khác của quốc gia	Vấn đề này bị loại bỏ khi sử dụng SUA
Độ phức tạp khi thực hiện	M3UA cần các dịch vụ của SCCP	Một lớp giao thức bị loại bỏ. Nhờ loại bỏ SCCP nên đã giảm độ phức tạp của nút mạng (về khía cạnh thực hiện cũng như quản lý), do đó giảm chi phí
Các khía cạnh định tuyến	M3UA xử lý bản tin từ mã điểm đến mã điểm (point code to point code)	SUA cho phép mạng IP sử dụng thông tin tiêu đề toàn cầu để định tuyến các bản tin
Các khía cạnh về địa chỉ	Sử dụng M3UA mỗi nút IP phải có đồng thời địa chỉ IP và mã điểm	Sử dụng SUA mỗi nút IP không cần dùng đến nguồn tài nguyên mã điểm đang khan hiếm
Các dịch vụ ISUP	Hỗ trợ	Có thể không hỗ trợ

Như chúng ta thấy, SUA cung cấp giải pháp tốt hơn so với M3UA. Nó rất thích hợp với việc thực hiện báo hiệu trong mạng IP toàn phần khi so sánh với SCCP hoặc M3UA. Khả năng mạnh mẽ về địa chỉ và định tuyến của SUA có thể làm giảm đáng kể trễ trong trao đổi thông tin báo hiệu.

Chương 8

CÁC ỨNG DỤNG CỦA CHUYỂN MẠCH MỀM TRONG MẠNG THẾ HỆ SAU

Như đã trình bày và phân tích kỹ trong các chương trước về các giải pháp công nghệ và kiến trúc hạ tầng mạng NGN dựa trên cơ sở mạng chuyển mạch gói tốc độ cao, công nghệ truyền dẫn WDM, SONET/SDH, IP/ATM, MPLS và chuyển mạch mềm đã tạo điều kiện cho sự hội tụ mạng và tích hợp dịch vụ, đảm bảo khả năng cung cấp nhiều dịch vụ với chất lượng cao, đem lại nhiều lợi ích cho cả khách hàng và cả nhà đầu tư, quản lý mạng và cung cấp dịch vụ. Trong những giải pháp xây dựng mạng NGN (cũng như mọi loại mạng truyền thông nào khác) thì chuyển mạch mềm (hay chuyển mạch nói chung) có vai trò quan trọng đặc biệt.

Trong chương này chúng ta sẽ xem xét các ứng dụng chính của chuyển mạch mềm trong mạng truyền thông. Những ứng dụng của chuyển mạch mềm sẽ được giới thiệu trong chương này bao gồm:

- Ứng dụng trong VoIP (Voice over IP)
- Ứng dụng làm SS7 PRI Gateway
- Ứng dụng trong tổng đài chuyển tiếp chuyển mạch gói (Packet Tandem)
- Ứng dụng tổng đài nội hạt

8.1 ỨNG DỤNG CHUYỂN MẠCH MỀM TRONG VoIP

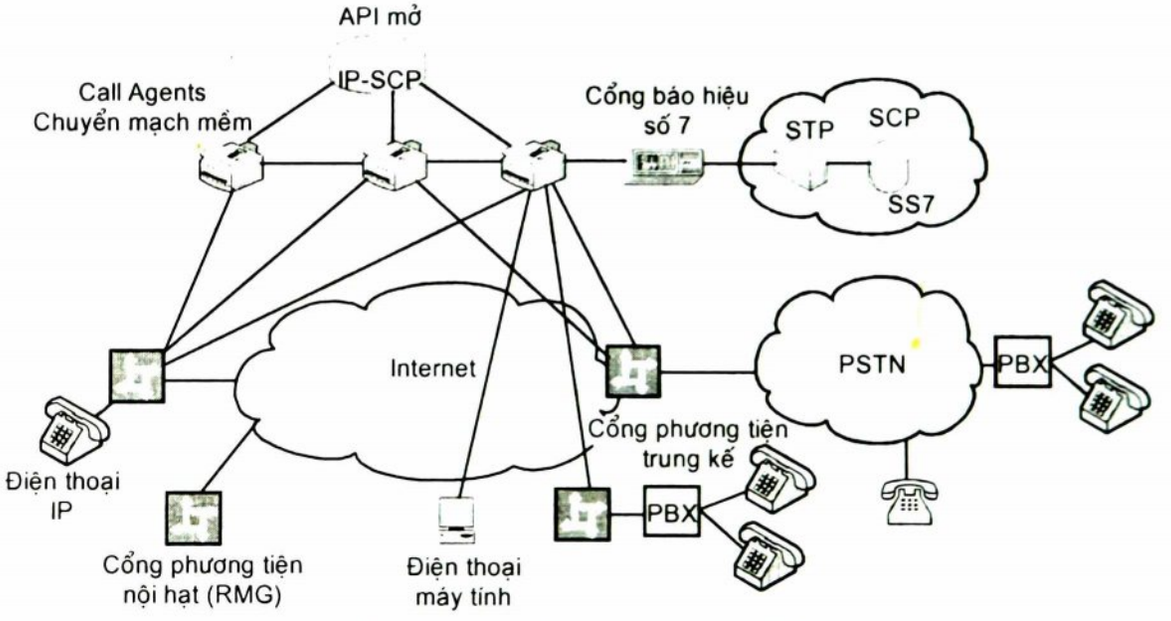
Có thể nói rằng chuyển mạch mềm trước hết sinh ra là để ứng dụng trong VoIP - công nghệ truyền tín hiệu điện thoại qua mạng IP. VoIP có thể hiểu một cách khái quát là công nghệ sử dụng các giao thức IP để phân phát các tín hiệu điện thoại số qua mạng.

Ngay từ khi phương thức chuyển mạch gói ra đời, với ưu điểm vượt trội về hiệu quả sử dụng tài nguyên mạng của nó so với phương thức chuyển mạch kênh nên các chuyên gia viễn thông rất quan tâm nghiên cứu việc truyền tín hiệu điện thoại qua mạng gói. Để gải quyết vấn đề này thì việc chế tạo được các bộ mã hóa/giải mã (codec) thoại đáp ứng các yêu cầu truyền thoại gói có ý nghĩa quan trọng đặc biệt. Nhờ những thành tựu mới hiện đại của công nghệ xử lý số tín hiệu, ngày nay đã chế tạo được nhiều kiểu Codec thoại sử dụng tốt cho VoIP.

So với điện thoại truyền thống, VoIP có bốn lợi ích vượt trội sau:

- Giảm giá thành liên lạc đường dài
- Số cuộc gọi nhiều hơn với băng thông nhỏ hơn
- Các dịch vụ gia tăng nhiều hơn và tốt hơn
- Sử dụng IP hiệu quả hơn

Hình 8.1 minh họa một phương án kiến trúc tổng quan của hệ thống VoIP. Trong kiến trúc này các MG được đặt ở cạnh của mạng Internet. MGC thực hiện chức năng xử lý và quản lý các cuộc gọi và nó đáp ứng quá trình báo hiệu của cuộc gọi. Nếu một MGC nào đó bị sự cố vì bất cứ lý do gì có thể thì một MGC khác sẽ sẵng sàng thay thế để đảm bảo tiếp tục duy trì hoạt động chức năng bình thường của mạng. Trong kiến trúc này thể hiện rõ rằng trí tuệ của các chức năng xử lý và điều khiển cuộc gọi đã được tách khỏi phần cứng MG để phân tán trên toàn mạng tại các MGC.



Hình 8.1 Kiến trúc mạng VoIP

Có một số kiểu MG khác nhau sử dụng trong kiến trúc này như: SOHO, Reisidential MG (cổng phương tiện nội hạt), Trunk MG (cổng trung kế) và Signalling Gateway (cổng báo hiệu). Bạn đọc quan tâm cấu tạo và chức năng của các kiểu MG có thể tham khảo chương 3.

Về các giao thức chính được sử dụng trong mạng VoIP gồm có: T.120, H.323, SIP, SGCP, MGCP... và các giao thức truyền tải IP khác như RTP/RTCP, UDP...

8.2 ỨNG DỤNG CHUYỂN MẠCH MỀM LÀM SS7 PRI GATEWAY

Ứng dụng này nhằm vào các nhà khai thác dịch vụ thoại, những doanh nghiệp đang tìm kiếm một giải pháp giá thành thấp thay cho chuyển mạch kênh truyền thống

để cung cấp giao diện tốc độ sơ cấp PRI (Primery Rate Interface) cho các nhà cung cấp dịch vụ Internet (ISP) phục vụ các đường truy nhập đường điện thoại (Dial-up).

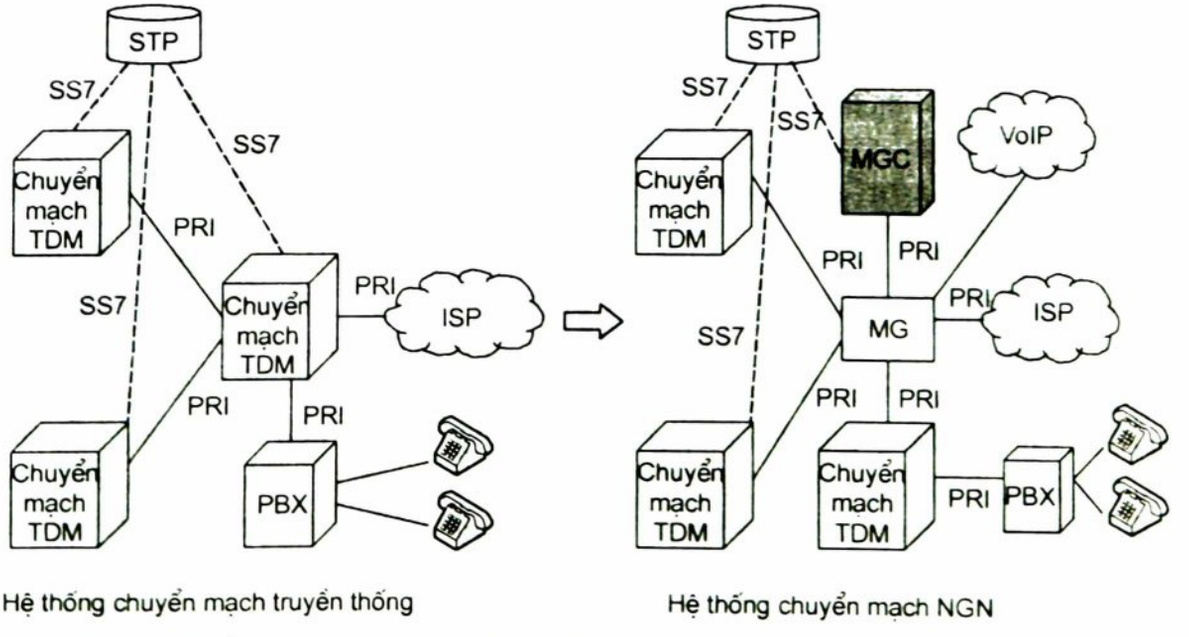
Sự bùng nổ về truy cập Internet qua đường điện thoại và khuynh hướng của các ISP muốn kết nối các Modem Server của họ với các luồng PRI làm cho các nhà cung cấp dịch vụ nhanh chóng hết cổng PRI hiện có. Hơn thế nữa, các kênh PRI do các ISP thuê thường mang lại cho nhà khai thác tổng đài lợi nhuận ít hơn so với các kênh PRI khác.

Bên cạnh việc thiếu các kênh PRI, lưu lượng truy cập Internet qua đường điện thoại làm quá tải và tắc nghẽn cho mạng chuyển mạch kênh. Bởi vì chuyển mạch kênh vốn được thiết kế để phục vụ các cuộc gọi có độ dài trung bình ngắn, nên khi khoảng thời gian trung bình tăng thêm do truy cập Internet, có xu hướng làm suy giảm tài nguyên tổng đài. Để duy trì chất lượng thoại cho các khách hàng sử dụng dịch vụ điện thoại thực sự, các nhà khai thác phải chọn một trong hai phương án:

1. Lắp đặt thêm tổng đài hoặc
2. Cung cấp cho các ISP các kênh PRI có lưu lượng tải thấp.

Cả hai phương án này đều tương đương nhau về mặt đầu tư.

Phần bên trái trong hình 8.2 minh họa mô hình mạng hiện nay của các nhà khai thác tổng đài nội hạt, nó cho thấy các kênh PRI phục vụ thông tin thoại thông thường và phục vụ các ISP là như nhau. Bởi vì phần lớn thuê bao Internet nằm ở phía thiết bị của nhà khai thác cấp cao hơn nên phần lớn lưu lượng số liệu từ mô-đem sẽ đi qua các kênh kết nối giữa thiết bị của nhà khai thác cấp cao và nhà khai thác cạnh tranh, hơn thế nữa không có sự phân biệt giữa lưu lượng thoại và lưu lượng số liệu Internet, điều đó dẫn đến tình trạng chuyển mạch của nhà khai thác cạnh tranh trở thành một "nút cổ chai" trên mạng.



Hình 8.2: Ứng dụng làm SS7 PRI Gateway của chuyển mạch mềm

Mô-đem vẫn sẽ là phương tiện thông dụng nhất để kết nối Internet trong một thời gian nữa, thực tế đó đòi hỏi các nhà khai thác tìm ra một giải pháp kinh tế cung cấp kênh PRI cho các ISP và chuyển các kênh PRI đang dùng cho các khách hàng điện thoại truyền thống.

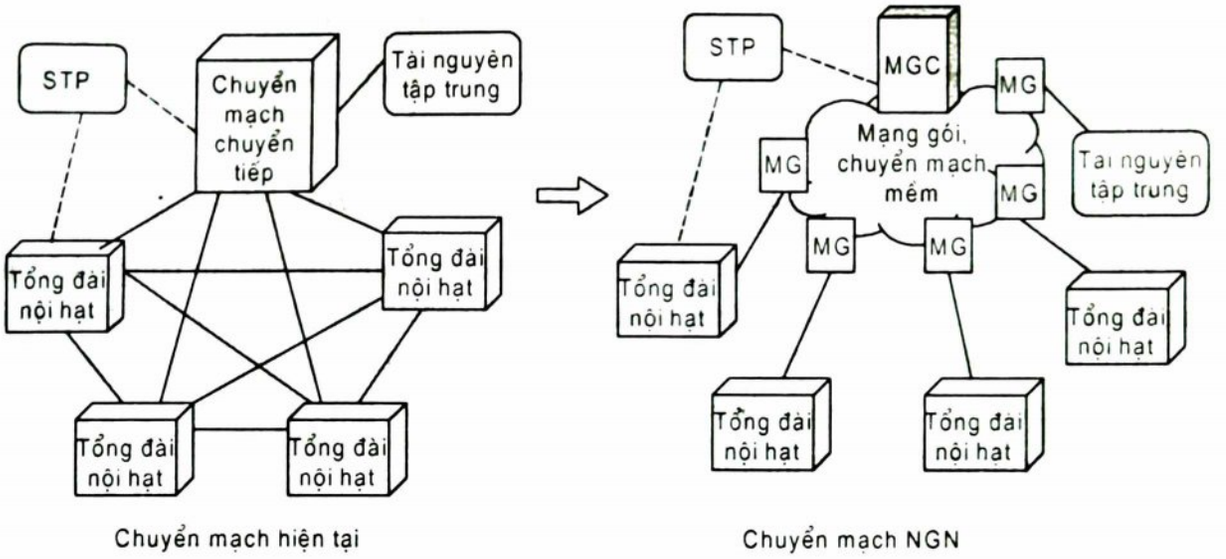
Ứng dụng chuyển mạch mềm làm SS7 PRI Gateway là một trong những giải pháp trong tình huống này. Như phần bên phải của hình 8.2 trên cho thấy, MGC và MG được đặt ở trung kế liên tổng đài giữa nhà khai thác cấp cao và nhà khai thác cạnh tranh. Chuyển mạch kênh kết nối với MG bằng giao diện TDM chuẩn còn liên lạc với MGC thông qua báo hiệu SS7. Các mô-đem server của ISP vì thế sẽ được chuyển sang kết nối với Media Gateway, giải phóng các luồng PRI cho chuyển mạch kênh TDM truyền thống. Khi một cuộc gọi Internet (Dial-up) hướng tới ISP từ phía tổng đài cấp cao, nó sẽ đi qua trung kế tới MG rồi được định hướng trực tiếp tới mô-đem server mà không qua chuyển mạch kênh như trước. Các cuộc gọi thoại vẫn diễn ra như bình thường.

Bên cạnh việc cung cấp các kênh PRI giá thành thấp, chịu được các cuộc gọi thời gian trung bình lâu hơn so với trước đây, ứng dụng SS7 PRI Gateway còn có khả năng cung cấp các dịch vụ mới như VoIP.

8.3 ỨNG DỤNG CHUYỂN MẠCH MỀM TRONG TỔNG ĐÀI CHUYỂN TIẾP GÓI (PACKET TANDEM)

Ứng dụng chuyển mạch mềm vào tổng đài chuyển tiếp gói (Packet Tandem) hướng vào các nhà cung cấp dịch vụ thoại truyền thống với mong muốn giảm vốn đầu tư và chi phí điều hành các tổng đài chuyển tiếp chuyển mạch kênh hiện nay, ngoài ra còn cung cấp các dịch vụ mới về số liệu. Giải pháp chuyển mạch TDM hiện nay đang bộc lộ dần nhược điểm trước nhu cầu ngày càng tăng nhưng rất thất thường của lưu lượng thông tin thoại nội hạt (phát sinh do truy cập Internet), vô tuyến và đường dài.

Phần bên trái của hình 8.3 cho thấy một mạng tổng đài TDM cấp thấp nhất (Cấp 5, tổng đài nội hạt, MSC của mạng di động...) được nối với nhau và nối tới tổng đài chuyển tiếp cấp cao hơn (tổng đài cấp 3, 4) bằng một mạng lưới trung kế điểm-điểm khá phức tạp. Khi một cuộc gọi diễn ra giữa hai tổng đài cấp thấp, thông tin sẽ đi trên trung kế nối trực tiếp giữa hai tổng đài, nếu đường nối trực tiếp đã sử dụng hết, cuộc gọi có thể được định tuyến thông qua tổng đài chuyển tiếp. Một số cuộc gọi (ví dụ như truy nhập hộp thư thoại hay quay số bằng giọng nói...) lại được định tuyến trực tiếp tới tổng đài chuyển tiếp để sử dụng các tài nguyên tập trung phục vụ cho các dịch vụ cao cấp. Kiến trúc này đã được sử dụng nhiều năm nay và cũng đã được cải tiến rất nhiều nhằm phục vụ các ứng dụng thoại, tuy nhiên vẫn có một số giới hạn như sau:



Hình 8.3: Ứng dụng trong tổng đài chuyển tiếp gói

• Chi phí điều hành và bảo dưỡng cao, mất thời gian; việc định lại cấu hình và nâng cấp mạng lưới phải tiến hành liên tục nhằm tránh bị nghẽn mạng, hơn nữa luôn phải thiết lập mạng lớn hơn nhu cầu thực tế cho các tổng đài chuyển tiếp. Ví dụ, khi một tổng đài nội hạt được thêm vào mạng lưới, phải xây dựng các nhóm trung kế từ tổng đài đó tới tổng đài chuyển tiếp và tới một số tổng đài nội hạt khác.

• Các trung kế điểm-điểm hoạt động với hiệu suất không cao vì chúng được thiết kế để hoạt động được trong những giờ cao điểm, và những giờ cao điểm này lại khác nhau trong các vùng của mạng (ví dụ ở khu doanh nghiệp là ban ngày còn ở khu dân cư lại là buổi đêm).

• Nếu có nhiều tổng đài chuyển tiếp trong mạng, mỗi tổng đài đó lại nối với một nhóm các tổng đài nội hạt, cuộc gọi có thể phải chuyển qua nhiều tổng đài chuyển tiếp để đến được nơi lưu giữ tài nguyên mạng (như trong trường hợp dịch vụ hộp thư thoại).

Chuyển mạch mềm chính là giải pháp cho vấn đề trên. Trong hình 8.3, phía bên phải cho thấy MGC cùng với các MG thay thế chức năng của các tổng đài chuyển tiếp chuyển mạch kênh trước đây, các tổng đài nội hạt kết nối tới các MG bằng giao diện chuẩn TDM thông thường và với MGC bằng báo hiệu số 7.

Mô hình này mang lại một số lợi ích so với mô hình mạng chuyển mạch kênh:

• Loại bỏ lưới trung kế hoạt động hiệu suất không cao, thay thế chúng bằng các "siêu xa lộ" trong mạng IP/ATM, SDH phục vụ cho các cuộc gọi cần chuyển tiếp, giảm tải cho các tổng đài chuyển tiếp truyền thống hoặc loại bỏ chúng hoàn toàn.

• Giảm được chi phí vận hành vì giảm được số tổng đài chuyển tiếp, số trung kế ít hơn (so với một mạng lưới trước đây), tránh không phải thiết kế các mạch TDM phức tạp.

• Giảm được một số lượng các cổng chuyển mạch dùng cho các trung kế giữa các tổng đài nội hạt với nhau.

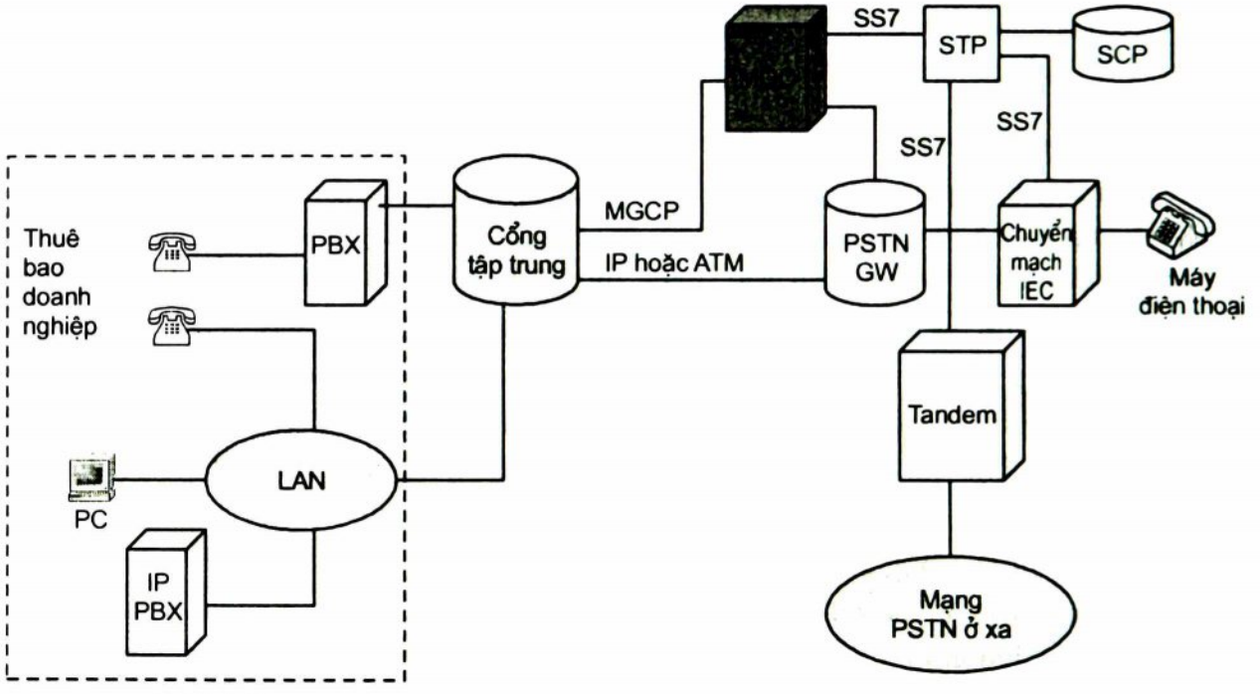
• Truy nhập các tài nguyên tập trung một cách hiệu quả hơn.

• Hợp nhất thông tin thoại và số liệu vào một mạng duy nhất, qua đó giảm vốn đầu tư và chi phí so với các mạng riêng biệt hiện nay cho thoại và số liệu.

8.4 ỨNG DỤNG CHUYỂN MẠCH MỀM TRONG TỔNG ĐÀI NỘI HẠT

Các nhà cung cấp dịch vụ nội hạt có thể triển khai mạng thế hệ sau bằng cách sử dụng MGC và các MG. Ngoài những chức năng như SS7 PRI Gateway hay tổng đài chuyển tiếp gói mà chúng ta đã đề cập đến ở phần trên MGC có thể đảm nhiệm chức năng của tổng đài nội hạt, cung cấp các dịch vụ thoại, gói và Internet.

a) Cung cấp dịch vụ cho doanh nghiệp



Chú thích:

PBX	Private Branch eXchange (Tổng đài nhánh)
PSTN-GW	PSTN Gateway (Cổng PSTN)
MGCP	MGC Protocol (Giao thức MGC)
ILEC-Sw	Incumbent Local Ex-Carrier Sw (Công ty điện thoại nội hạt truyền thống)
STP	Signal Transfer Point (Điểm chuyển tiếp tín hiệu)
SCP	Signal Control Point (Điểm điều khiển tín hiệu)

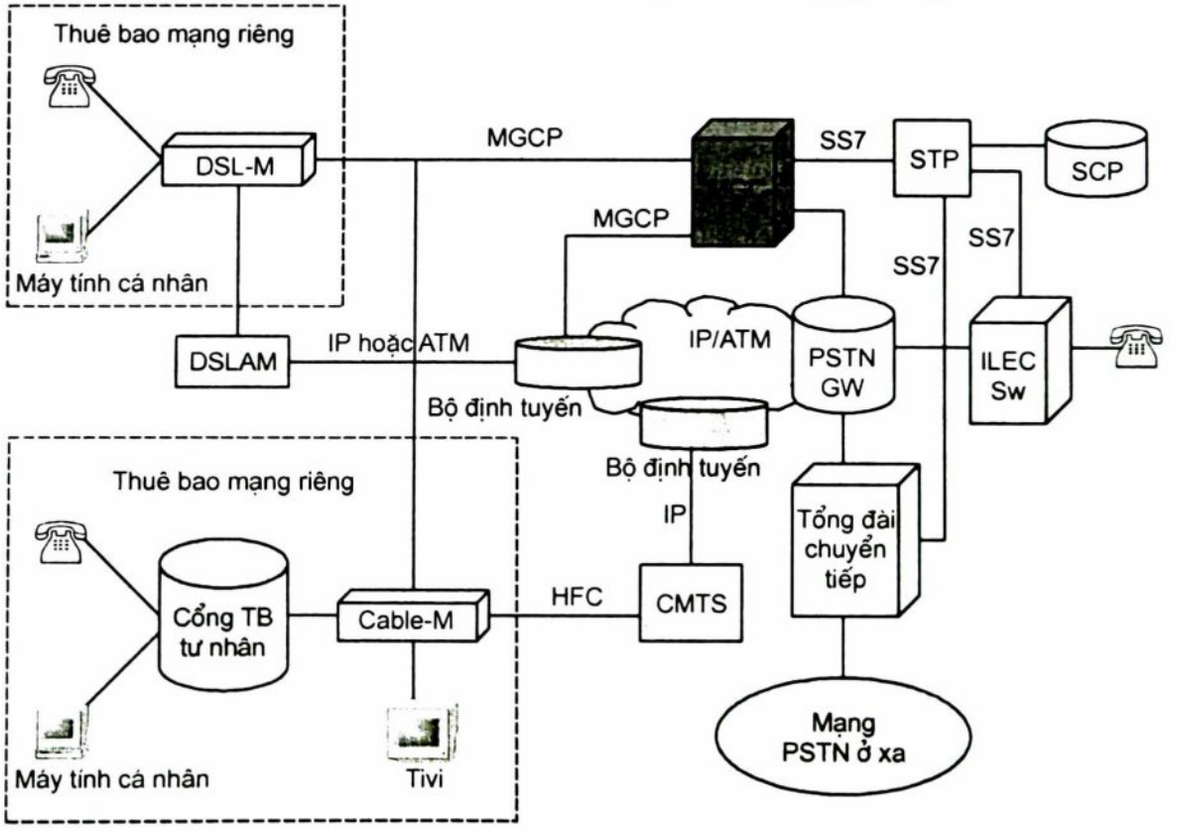
Hình 8.4: Mạng thế hệ sau và thuê bao doanh nghiệp

Một doanh nghiệp vừa hoặc nhỏ thông thường sử dụng 8 đến 16 đường dây thoại và một kết nối Internet. Trong mạng thế hệ mới băng thông sẽ được chia sẻ mềm dẻo giữa các dịch vụ thoại và dịch vụ dữ liệu. Hình 8.4, ta thấy thuê bao doanh nghiệp có thể sử dụng dịch vụ PBX truyền thống (chỉ cho thoại) hoặc có thể sử dụng một kết nối dữ liệu duy nhất từ cổng truy nhập tới mạng LAN và sử dụng kết nối này cho cả các dịch vụ thoại và dữ liệu.

Chú ý rằng có thể sử dụng lại hệ thống cáp đã có sẵn và qua đường dữ liệu này thuê bao có thể kết nối với các mạng LAN khác, với mạng WAN, mạng ATM hoặc mạng của các nhà cung cấp dịch vụ Internet. Do phần lớn các sản phẩm chuyển mạch mềm tách riêng biệt 2 mô-đun phần mềm giám soát cuộc gọi và xử lý tính năng nên có thể cung cấp nhiều dịch vụ Centrex (một dịch vụ do các công ty thoại nội hạt cung cấp) như làm việc tại nhà và nhân viên di động. Như vậy việc tổ chức một văn phòng ảo là hoàn toàn có thể thực hiện được.

b) Cung cấp dịch vụ cho mạng riêng

Dĩ nhiên các thuê bao tư nhân có thể chỉ sử dụng dịch vụ thoại như trước đây. Ngoài ra nếu sử dụng mô-đem cáp hoặc xDSL thì thuê bao có thể sử dụng các dịch vụ số liệu. Hình 8.5 minh họa cấu trúc và các thành phần của mạng riêng.



Chú thích:

DSL-M	DSL Modem	Mô-đem DSL
DSLAM	DSL Access Multiplex	Bộ ghép kênh truy nhập đường dây thuê bao số
Cable-M	Cable Modem	Mô-đem cáp
HFC	Hybrid Fiber Cable	Cáp quang hỗn hợp
CMTS	Cable Modem Transmission Switch	Chuyển mạch truyền dẫn mô-đem cáp
PSTN-GW	PSTN Gateway	Cổng PSTN
STP	Signal Transfer Point	Điểm chuyển tiếp tín hiệu
SCP	Signal Control Point	Điểm điều khiển tín hiệu
MGCP	MGC Protocol	Giao thức MGC
ILEC-Sw	Incumbent Local Exchange Carrier Switch	Chuyển mạch tổng đài nội hạt chủ đạo

Hình 8.5: Mạng thế hệ sau và thuê bao mạng riêng

THUẬT NGỮ VIẾT TẮT

AAA	Authentication, Authorization and Accounting	Chủ quyền, xác minh và tính cước
ABNF	Augumented Backus - Naur Form	Chuẩn mã hoá văn bản
A-F	Acouting - Function	Chức năng tính cước
AG	Access Gateway	Cổng truy nhập
AGS-F	Access Gateway Signaling - Function	Chức năng báo hiệu cổng truy nhập
AIN	Advanced Intelligent Network	Mạng thông minh tiên tiến
API	Application Programming Interface	Giao diện lập trình ứng dụng
AS	Application Server	Máy chủ ứng dụng
AS-F	AS-Function	Chức năng máy chủ ứng dụng
ASN no1	Abstraction Syntax Notation 1	Chuẩn mã hoá nhị phân
ATM	Asynchronous Transfer Mode	Chế độ chuyển giao không đồng bộ
BAN	Broadband Access Network	Mạng truy nhập băng rộng
BSSAP	Base Station Subsystem Application Part	Phần ứng dụng phân hệ trạm gốc của hệ thống GSM
CA	Call Agent	Tác nhân cuộc gọi
CAD	Computer Aided Designing	Thiết kế được hỗ trợ bằng máy tính
CAM	Computer Aided Manufactoring	Sản xuất được hỗ trợ bằng máy tính
CAMEL	Customered Application for Mobile Network Enhanced Logic	Lô-gíc cao cấp của những ứng dụng theo yêu cầu khách hàng cho mạng di động
CAPEX	CAPital EXpense	Chi phí vốn
CATV	Cable TV	Truyền hình cáp
CCS7	Common Chanel Signalling number 7	Báo hiệu kênh chung số 7
DTMF	Dual Tone Multi Frequency	Âm báo mã đa tần
DSS1	Digital Signalling System number 1	Hệ thống báo hiệu thuê bao số số 1
FRS	Frame Relay Switch	Chuyển mạch chuyển tiếp khung
FS	Feature Server	Máy chủ chức năng
GSM	Global System for Mobile Telecommunication	Hệ thống thông tin di động toàn cầu
HTML	Hyper Text Markup Language	Ngôn ngữ đánh dấu siêu văn bản
HTTP	Hyper Text Transport Protocol	Giao thức truyền tải siêu văn bản
IAD	Integrated Access Device	Thiết bị truy nhập tích hợp
IETF	Internet Engineering Task Force	Nhóm đặc trách kỹ thuật Internet
IN	Intelligent Network	Mạng thông minh
INAP	IN Application Protocol	Giao thức ứng dụng mạng IN

IP	Internet Protocol	Giao thức Internet
IPDC	IP Device Control	Điều khiển thiết bị IP
ISC	International Softswitch Consortium	Tổ chức chuyển mạch mềm quốc tế
ISDN	Integrated Service Digital Network	Mạng số liên kết đa dịch vụ
ISP	Internet Service Provider	Nhà cung cấp dịch vụ Internet
ISUP	ISDN User Part	Phần ứng dụng dành cho người dùng ISDN
ITU, ITU-T	International Telecommunication Union, ITU - Telecom Sector	Liên minh Viễn thông Quốc tế, bộ phận tiêu chuẩn hóa viễn thông của ITU
IUA	ISDN User Adaptation	Thích ứng người dùng ISDN
IW-F	Interworking- Function	Chức năng liên kết mạng
JAIN	Java Application Interface Network	Mạng giao diện ứng dụng với ngôn ngữ Java
LAN	Local Area Network	Mạng cục bộ
LDAP	Lightweight Directory Access Protocol	Giao thức truy nhập thư mục trọng lượng thấp
LSI/VLSI	Large Scale Integrated/Very Large Scale Integrated circuit	Vi mạch cỡ lớn và cực lớn
MAP	Mobile Application Part	Phần ứng dụng di động
M2UA	MTP level 2 User Adaptaion	Tương thích với người dùng mức 2
MCU	Multipoint Control Unit	Khối điều khiển đa điểm
MDCP	Media Device Control Protocol	Giao thức điều khiển thiết bị phương tiện
MEGACO	MEdia GAteway COntroller	Giao thức điều khiển cổng phương tiện
MG	Media Gateway	Cổng phương tiện
MGC	Media Gateway Controler	Bộ điều khiển cổng phương tiện
MGC-F	MGC- Function	Chức năng MGC
MGCP	Media Gateway Control Protocol	Giao thức điều khiển cổng lưu lượng
MG-F	MG-Function	Chức năng cổng MG
MHS	Message Handling Service	Dịch vụ xử lý bản tin
MPLS	MultiProtocol Label Switching	Chuyển mạch nhãn đa giao thức
NGN	Next Generation Network	Mạng thế hệ sau
NIF	Nodal Interworking Function	Chức năng liên mạng của nút
NNI	Network-Network Interface	Giao diện mạng-mạng
OPEX	OPeration EXpense	Chi phí vận hành mạng
OSS	Operational Support System	Hệ thống hỗ trợ vận hành
PBX	Private Branch eXchange	Tổng đài nhánh
PDU	Protocol Datagram Unit	Đơn vị gói giao thức
PCM	Pulse Code Mudulation	Điều chế xung mã
PLMN	Public Land Mobile Network	Mạng di động mặt đất công cộng
PRI	Primary Rate Interface	Giao diện tốc độ sơ cấp

PSDN	Public Switched Data Network	Mạng dữ liệu chuyển mạch công cộng
PSTN	Public Switched Telephone Network	Mạng điện thoại chuyển mạch công cộng
QoS	Quality of Service	Chất lượng dịch vụ
RAN	Radio Access Network	Mạng truy nhập vô tuyến
RANAP	Radio Access Network Application Part	Phần ứng dụng mạng truy nhập vô tuyến
R-F	Routing- Function	Chức năng định tuyến
RFC	Request For Comment	Yêu cầu dẫn chú
RGW	Residential Gateway	Cổng nội hạt
RTCP	Real Time Transport Control Protocol	Giao thức điều khiển truyền tải thời gian thực
RTP	Real Time Transport Protocol	Giao thức truyền tải thời gian thực
SPC	Stored Programme Control	Điều khiển theo chương trình lưu trữ
SCTP	Stream Control Transport Protocol	Giao thức truyền tải điều khiển dòng
SDP	Session Discription Protocol	Giao thức mô tả phiên
SG	Signalling Gateway	Cổng báo hiệu
SGCP	Simple Gateway Control Protocol	Giao thức điều khiển cổng đơn giản
SIP	Session Intiation Protocol	Giao thức khởi đầu phiên
SIP-T	Session Intiation Protocol for Telephony	Phần mở rộng giao thức SIP dành cho thoại
SNMP	Simple Network Management Protocol	Giao thức quản lý mạng đơn giản
SS7	Signaling System No7	Hệ thống báo hiệu số 7
STP	Signal Transfer Point	Điểm truyền tải báo hiệu
SUA	SCCP User Adatation	Thích ứng người dùng SCCP
TCP	Transmission Control Protocol	Giao thức điều khiển truyền tải
TDM	Time Division Mutiplexing	Ghép kênh theo thời gian
TGW	Trunk GateWay	Cổng trung kế
UAC	User Agent Client	Máy trạm tác nhân người dùng
UAS	User Agent Server	Máy chủ tác nhân người dùng
UDP	User Datagram Protocol	Giao thức gói tin người dùng
URI	Uniform Resoure Identifier	Nhận dạng tài nguyên đồng nhất
URL	Uniform Resoure Locator	Bộ định vị tài nguyên đồng nhất
VoIP	Voice over IP	Thoại qua mạng IP
WAN	Wide Area Network	Mạng diện rộng
WDM	Wavelenght Division Multiplexing	Ghép kênh phân chia theo bước sóng

TÀI LIỆU THAM KHẢO

1. *IP Switching Protocols and Architectures*, Christopher Y.Metz. Mc Graw-Hill 1999
2. *Information about NGN*, www.de.infowin/ACTS/ANALYSYS/PRODUCTS/MATHEMTIC/NGI/ch5/ch5.pdf
3. *Softswitch: Next Generation Telecommunication Switching Platform*, Sun Microsystems, www.sun.com
4. *Voice over IP Protocols*, www.Vivoda.org
5. *SIP infrastructure for NGN*, www.hssworld.com
6. *Acceletating the Deployment of VoIP and VoATM*, www.ec.org
7. *Local exchange Softswitch System: Softswitch and Packet Voice*, www.ec.org
8. *Reference Architecture - Softswitch Consortium*, ISC June-2002
9. *SIP Tutorial*, www.forkus.gmd.de/mobis/siptutorial
10. *Standard of SIP*, *rfc 2543-SIP*, IETF-RFC-2543, tháng 3/1999
11. Một số địa chỉ Internet khác: www.ietf.org, www.itu.int, www.packetizer.com

MỤC LỤC

Lời nói đầu 5

Chương 1: TỔNG QUAN VỀ HIỆN TRẠNG VÀ XU HƯỚNG PHÁT TRIỂN CỦA VIỄN THÔNG VÀ CÔNG NGHỆ THÔNG TIN 7

1.1 Sự phát triển của kỹ thuật viễn thông và máy tính 7

1.1.1 Viễn thông 7

1.1.2 Máy tính 11

1.2 Xu hướng hội tụ công nghệ và tích hợp dịch vụ 14

1.3 Các dịch vụ viễn thông cơ bản 15

Chương 2: TỔNG QUAN VỀ MẠNG THẾ HỆ SAU VÀ CHUYỂN MẠCH MỀM 21

2.1 Cấu hình tổng quan mạng thế hệ sau 21

2.1.1 Sự ra đời của mạng thế hệ sau 21

2.1.2 Các đặc điểm và ưu điểm của mạng 22

2.1.3 Mô hình kiến trúc mạng 23

2.2 Các khái niệm chuyển mạch mềm 26

Chương 3: MÔ HÌNH THAM CHIẾU CHUYỂN MẠCH MỀM 29

3.1 Các mặt phẳng chức năng 29

3.1.1 Mặt phẳng truyền tải 29

3.1.2 Mặt phẳng báo hiệu và điều khiển cuộc gọi 30

3.1.3 Mặt phẳng ứng dụng và dịch vụ 31

3.1.4 Mặt phẳng quản lý và bảo dưỡng mạng 31

3.2 Các thực thể chức năng 31

3.2.1 Chức năng điều khiển cổng phương tiện (MGC-F) 32

3.2.2 Chức năng định tuyến cuộc gọi và tính cước (R-F, A-F) 32

3.2.3 Chức năng cổng báo hiệu và báo hiệu cổng truy nhập (SG-F/AGS-F) 33

3.2.4 Chức năng máy chủ ứng dụng (AS-F) 33

3.2.5 Chức năng cổng phương tiện (MG-F) 34
3.2.6 Chức năng máy chủ phương tiện (MS-F) 35
3.3 So sánh chuyển mạch mềm và chuyển mạch truyền thống 36
3.3.1 Kiến trúc các hệ thống chuyển mạch 36
3.3.2 Phương thức xử lý cuộc gọi 38
3.3.3 Các ưu điểm của chuyển mạch mềm 39
Chương 4: CÁC GIAO THỨC CƠ BẢN CỦA CHUYỂN MẠCH MỀM 43
4.1 Tổng quan về báo hiệu trong chuyển mạch mềm 43
4.2 Giao thức H.323 45
4.3 Giao thức SIP 48
4.4 Giao thức MGCP-MEGACO/H.248 50
4.5 Giao thức SIGTRAN 51
Chương 5: GIAO THỨC KHỞI TẠO PHIÊN SIP 53
5.1 Định nghĩa và các thành phần của hệ thống SIP 53
5.1.1 Các định nghĩa và khái niệm cơ bản 53
5.1.2 Các thành phần của kiến trúc SIP 54
5.2 Chức năng và tính năng của SIP 56
5.2.1 Các chức năng của SIP 56
5.2.2 Các tính năng của SIP 57
5.3 Hoạt động của SIP 58
5.3.1 Địa chỉ SIP (SIP Address) 58
5.3.2 Giao dịch SIP (SIP Transaction) 61
5.3.3 Lời mời SIP 62
5.3.4 Quá trình định vị tới máy chủ SIP 63
5.3.5 Định vị người dùng 64
5.3.6 Thay đổi một phiên hiện tại 65
5.3.7 Thiết lập và giải phóng cuộc gọi SIP 65
5.4 Bản tin SIP 66
5.4.1 Bản tin yêu cầu (Request) 66

5.4.2 Bản tin đáp ứng (Response) 72
5.4.3 Trường tiêu đề trong bản tin SIP 74
5.4.4 Mã đáp ứng của bản tin SIP 86
5.5 Thân bản tin SIP 91
5.5.1 Thành phần trong thân bản tin 91
5.5.2 Kiểu và độ dài thân bản tin 91
5.6 Khuôn dạng bản tin SIP 92
5.7 Hoạt động của các thành phần SIP 92
5.7.1 Hoạt động của SIP Client và SIP Server 92
5.7.2 Hoạt động của Proxy Server, Location Server và Redirect Server 95
5.7.3 Hoạt động của UA (User Agent) 96
Chương 6: GIAO THỨC ĐIỀU KHIỂN CỔNG PHƯƠNG TIỆN - MEGACO/H.248 98
6.1 Giới thiệu chung về giao thức điều khiển cổng phương tiện. 98
6.2 Giao thức MEGACO/H.248 99
6.2.1 Tổng quan về giao thức MEGACO/H.248 99
6.2.2 Chức năng của giao thức MEGACO/H.248 100
6.2.3 Vị trí của giao thức MEGACO/H.248 trong mô hình OSI 100
6.2.4 Các khái niệm trong giao thức MEGACO/H.248 101
6.2.5 Truyền dẫn các bản tin của giao thức MEGACO/H.248 104
6.2.6 Các lệnh của thức MEGACO/H.248 105
6.2.7 Mã hóa lệnh của giao thức MEGACO/H.248 105
6.2.8 Cú pháp lệnh của giao thức MEGACO/H.248 106
6.2.9 Cấu trúc bản tin MEGACO/H.248 108
6.3 Hoạt động của giao thức MEGACO/H.248 109
6.4 Các ưu điểm của MEGACO/H.248 so với các giao thức điều khiển cổng phương tiện khác 110
6.5 Kết nối giữa hai cổng nội hạt (RGW) 111
6.6 Kết nối từ RGW tới TGW 124
6.7 Kết nối từ SS7-TGW tới RGW 127
6.8 MEGACO/H.248 hỗ trợ các dịch vụ bổ sung 129

Chương 7: GIAO THỨC TRUYỀN TẢI BÁO HIỆU SIGTRAN 141
7.1 Mô hình kiến trúc mạng tích hợp 141
7.2 Tổng quan về giao thức SIGTRAN 144
7.3 Giao thức SCTP 146
7.3.1 Vấn đề đối với giao thức TCP 146
7.3.2 Giao thức truyền dẫn điều khiển luồng SCTP 147
7.3.3 M2UA 153
7.3.4 M2PA 156
7.3.5 So sánh M2UA và M2PA 158
7.3.6 M3UA 158
7.3.7 SUA 160
7.3.8 So sánh M3UA và SUA 162
Chương 8: CÁC ỨNG DỤNG CỦA CHUYỂN MẠCH MỀM TRONG MẠNG THẾ HỆ SAU 163
8.1 Ứng dụng chuyển mạch mềm trong VoIP 163
8.2 Ứng dụng chuyển mạch mềm làm SS7 PRI Gateway 164
8.3 Ứng dụng chuyển mạch mềm trong tổng đài chuyển tiếp gói (Packet Tandem) ... 166
8.4 Ứng dụng chuyển mạch mềm trong tổng đài nội hạt 168
Thuật ngữ viết tắt 171
Tài liệu tham khảo 174

CHUYỂN MẠCH MỀM
VÀ ỨNG DỤNG TRONG MẠNG VIỄN THÔNG THẾ HỆ SAU

Chịu trách nhiệm xuất bản
LƯU ĐỨC VĂN

Biên tập:	TRẦN VŨ THƯỞNG
	NGUYỄN VĂN VĨNH
Chế bản:	NGUYỄN MẠNH HOÀNG
Sửa bản in:	NGUYỄN VĂN VĨNH
Trình bày bìa:	PHAN THẾ VINH

NHÀ XUẤT BẢN BƯU ĐIỆN

Trụ sở: 18 Nguyễn Du, TP. Hà Nội
Điện thoại: 04-9431283, 9431284 Fax: 04-9431285
E-mail: bientap@hn.vnn.vn Website: www.nxbbuudien.com.vn

Chi nhánh TP. HCM: 27 Nguyễn Bỉnh Khiêm, Quận I, TP. Hồ Chí Minh
Điện thoại: 08-9100925 Fax: 08-9100924
E-mail: chinhanh-nxbbd@hcm.vnn.vn

Chi nhánh TP. Đà Nẵng: 42 Trần Quốc Toản, Quận Hải Châu, TP. Đà Nẵng
Điện thoại: 0511-897467 Fax: 0511-897467
E-mail: pnbich@mpt.gov.vn

In 800 bản, khổ 19x27cm tại Công ty Cổ phần in Bưu điện
Số đăng ký kế hoạch xuất bản: 96-2006/CXB/50-08/BuĐ
Số quyết định xuất bản: 54/QĐ-NXB BĐ ngày 11/5/2006
In xong và nộp lưu chiểu tháng 5 năm 2006.